

## SKRIPSI

PEMBANGKIT TIMELAPSE PENGEMBANGAN PROYEK  
PERANGKAT LUNAK BERBASIS WEB



Billy Adiwijaya

NPM: 2015730053

PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS  
UNIVERSITAS KATOLIK PARAHYANGAN  
2019



**UNDERGRADUATE THESIS**

**TIMELAPSE GENERATOR FOR WEB-BASED SOFTWARE  
PROJECT DEVELOPMENT**



**Billy Adiwijaya**

**NPM: 2015730053**

**DEPARTMENT OF INFORMATICS  
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES  
PARAHYANGAN CATHOLIC UNIVERSITY  
2019**



## ABSTRAK

Git merupakan perangkat lunak *Version Control Systems*. *Version Control* adalah sistem yang merekam perubahan pada *file* atau sekumpulan *file* dari waktu ke waktu. Perubahan yang terjadi pada repositori dicatat oleh Git dalam bentuk histori *commit*. Dengan adanya histori *commit*, perkembangan suatu proyek perangkat lunak dapat dipantau. Akan tetapi, untuk perangkat lunak yang memiliki banyak *commit*, pemantauan progres dapat memakan waktu yang lama. Oleh karena itu, dibuatlah perangkat lunak yang dapat membangkitkan animasi *timelapse* dari pengembangan proyek perangkat lunak.

Perangkat lunak pembangkit *timelapse* ini hanya akan berfokus pada pengembangan proyek perangkat lunak berbasis *web*. Animasi *timelapse* dibuat dengan cara menggabungkan *screenshot* halaman *web* pada setiap *commit* menjadi satu *file* bertipe GIF. Penelusuran *commit* dilakukan menggunakan JGit. *Screenshot* halaman *web* pada setiap *commit* diambil menggunakan Selenium WebDriver.

Pengujian dilakukan dengan menggunakan beberapa repositori dari situs web dan WebDriver. Berdasarkan hasil pengujian, program dapat bejalan dengan baik pada beberapa WebDriver. Pada EdgeDriver, tampilan *layout* dari halaman *web* menjadi tidak rapih. Program dapat berjalan dengan baik dan dapat membangkitkan animasi timelapse pada beberapa repositori dari situs web. Ukuran *file* hasil animasi bergantung pada banyaknya *commit* pada repositori. Semakin banyak jumlah *commit*, semakin besar ukuran *file*.

**Kata-kata kunci:** Git, JGit, Selenium WebDriver, timelapse, commit



## ABSTRACT

Git is a Version Control Systems. Version control is a system that records changes to a file or set of files over time. Changes in a repository are recorded by Git in the form of commit history. Software development can be tracked via commit history. However, it will take time to monitor the progress of software that has many commits. Therefore, software that can generate timelapse animation for software project development is made.

This software only focus on web-based software project development. Timelapse animation can be made by combining screenshots of web page into one GIF file. Commits can be traversed using JGit. Screenshot of web page in every commit can be taken using Selenium WebDriver.

Testing is done by trying to run this software using several websites and WebDriver. The result of software testing said that software can run well on several WebDriver. On EdgeDriver, webpage layout is not neat. Software can run well on several websites. The size of animation file depends on the number of commit in a repository. The more the number of commits made, the bigger the file size of the animation file gets.

**Keywords:** Git, JGit, Selenium WebDriver, timelapse, commit



# DAFTAR ISI

|   |             |
|---|-------------|
| <b>KATA PENGANTAR</b>                           | <b>xv</b>   |
| <b>DAFTAR ISI</b>                               | <b>xvii</b> |
| <b>DAFTAR GAMBAR</b>                            | <b>xix</b>  |
| <b>DAFTAR TABEL</b>                             | <b>xxi</b>  |
| <b>1 PENDAHULUAN</b>                            | <b>1</b>    |
| 1.1 Latar Belakang . . . . .                    | 1           |
| 1.2 Rumusan Masalah . . . . .                   | 1           |
| 1.3 Tujuan . . . . .                            | 2           |
| 1.4 Batasan Masalah . . . . .                   | 2           |
| 1.5 Metodologi . . . . .                        | 2           |
| 1.6 Sistematika Pembahasan . . . . .            | 2           |
| <b>2 LANDASAN TEORI</b>                         | <b>5</b>    |
| 2.1 Git . . . . .                               | 5           |
| 2.1.1 Version Control Systems . . . . .         | 5           |
| 2.1.2 Cara Kerja Git . . . . .                  | 7           |
| 2.1.3 Operasi Dasar pada Git . . . . .          | 10          |
| 2.1.4 Git Checkout . . . . .                    | 12          |
| 2.2 JGit . . . . .                              | 12          |
| 2.2.1 Kelas Repository . . . . .                | 12          |
| 2.2.2 Kelas Ref . . . . .                       | 13          |
| 2.2.3 Kelas FileRepository . . . . .            | 13          |
| 2.2.4 Kelas Git . . . . .                       | 13          |
| 2.2.5 Kelas CheckoutCommand . . . . .           | 15          |
| 2.2.6 Kelas ListBranchCommand . . . . .         | 15          |
| 2.2.7 Enum ListBranchCommand.ListMode . . . . . | 15          |
| 2.2.8 Kelas LogCommand . . . . .                | 15          |
| 2.2.9 Kelas ResetCommand . . . . .              | 15          |
| 2.2.10 Kelas ResetCommand.ResetType . . . . .   | 16          |
| 2.2.11 Kelas RevCommit . . . . .                | 16          |
| 2.2.12 Kelas PersonIdent . . . . .              | 17          |
| 2.3 Selenium WebDriver . . . . .                | 17          |
| 2.3.1 Interface WebDriver . . . . .             | 17          |
| 2.3.2 Interface WebDriver.Options . . . . .     | 18          |
| 2.3.3 Interface WebDriver.Window . . . . .      | 18          |
| 2.3.4 Kelas ChromeDriver . . . . .              | 18          |
| 2.3.5 Kelas FirefoxDriver . . . . .             | 19          |
| 2.3.6 Kelas OperaDriver . . . . .               | 19          |
| 2.3.7 Kelas SafariDriver . . . . .              | 19          |

|  |   |           |
|--|---|-----------|
| 2.3.8  | Kelas InternetExplorerDriver . . . . .                    | 19        |
| 2.3.9  | Kelas EdgeDriver . . . . .                                | 19        |
| 2.3.10   | Kelas HtmlUnitDriver . . . . .                            | 19        |
| 2.3.11   | Interface OutputType . . . . .                            | 19        |
| 2.3.12   | Interface TakesScreenshot . . . . .                       | 20        |
| 2.4  | Apache Commons CLI . . . . .                              | 20        |
| 2.4.1  | Interface CommandLineParser . . . . .                     | 20        |
| 2.4.2  | Kelas DefaultParser . . . . .                             | 20        |
| 2.4.3  | Kelas CommandLine . . . . .                               | 21        |
| 2.4.4  | Kelas Options . . . . .                                   | 21        |
| 2.4.5  | Kelas Option . . . . .                                    | 21        |
| 2.4.6  | Kelas Option.Builder . . . . .                            | 22        |
| <b>3</b>   | <b>ANALISIS</b>   | <b>25</b> |
| 3.1  | Analisis Aplikasi Sejenis . . . . .                       | 25        |
| 3.2  | Analisis Penggunaan JGit dan Selenium WebDriver . . . . . | 27        |
| 3.2.1  | Analisis Penggunaan JGit . . . . .                        | 27        |
| 3.2.2  | Analisis Penggunaan Selenium WebDriver . . . . .          | 28        |
| 3.3  | Analisis Fitur Aplikasi yang Dibangun . . . . .           | 29        |
| 3.4  | Prapengujian Website Piktora . . . . .                    | 32        |
| 3.4.1  | Perbedaan Letak <i>File</i> . . . . .                     | 34        |
| 3.4.2  | Permasalahan Konfigurasi <i>Database</i> . . . . .        | 35        |
| 3.4.3  | Permasalahan Migrasi <i>Database</i> . . . . .            | 35        |
| <b>4</b>   | <b>PERANCANGAN</b>  | <b>37</b> |
| 4.1  | Perancangan Kelas . . . . .                               | 37        |
| 4.2  | Perancangan Antarmuka . . . . .                           | 39        |
| <b>5</b>   | <b>IMPLEMENTASI DAN PENGUJIAN</b>                         | <b>43</b> |
| 5.1  | Implementasi . . . . .                                    | 43        |
| 5.1.1  | Lingkungan Implementasi . . . . .                         | 43        |
| 5.1.2  | Hasil Implementasi . . . . .                              | 43        |
| 5.2  | Pengujian . . . . .                                       | 45        |
| 5.2.1  | Pengujian Fungsional . . . . .                            | 45        |
| 5.2.2  | Pengujian Eksperimental . . . . .                         | 50        |
| <b>6</b>   | <b>KESIMPULAN DAN SARAN</b>                               | <b>61</b> |
| 6.1  | Kesimpulan . . . . .                                      | 61        |
| 6.2  | Saran . . . . .   | 61        |
| <b>DAFTAR REFERENSI</b>  | <b>63</b>   |           |
| <b>A KODE PROGRAM</b>  | <b>65</b>   |           |
| <b>B LIBRARY UNTUK MEMBUAT FILE GIF</b>                                    | <b>73</b>   |           |
| <b>C KODE PROGRAM PROYEK PIKTORA PADA COMMIT 89000BE (12 JANUARI 2018)</b> | <b>77</b>   |           |

## DAFTAR GAMBAR

|  |    |
|--|----|
| 2.1 Local version control[1]. . . . .  | 5  |
| 2.2 Centralized version control[1]. . . . .  | 6  |
| 2.3 Distributed version control[1]. . . . .  | 7  |
| 2.4 Menyimpan data sebagai <i>snapshots</i> dari <i>project</i> [1]. . . . .   | 8  |
| 2.5 Menyimpan data sebagai perubahan terhadap versi dasar dari setiap <i>file</i> [1]. . . . .                                       | 8  |
| 2.6 <i>Working tree</i> , <i>Staging area</i> , dan Git direktori[1]. . . . .  | 9  |
| 2.7 <i>Checkout</i> pada <i>commit</i> . . . . .   | 12 |
| 3.1 Visualisasi proyek perangkat lunak menggunakan Gource. . . . .   | 25 |
| 3.2 Histori <i>commit</i> direpresentasikan sebagai Directed Acyclic Graph. . . . .  | 28 |
| 3.3 <i>Browser</i> yang dikontrol oleh ChromeDriver. . . . .   | 29 |
| 3.4 <i>Use case diagram</i> perangkat lunak. . . . .   | 31 |
| 4.1 Diagram kelas. . . . .   | 37 |
| 4.2 Rancangan <i>output</i> jika terdapat satu halaman <i>web</i> . . . . .  | 41 |
| 4.3 Rancangan <i>output</i> jika terdapat dua halaman <i>web</i> . . . . .   | 41 |
| 4.4 Rancangan <i>output</i> jika terdapat tiga halaman <i>web</i> . . . . .  | 42 |
| 4.5 Rancangan <i>output</i> jika terdapat empat halaman <i>web</i> . . . . .   | 42 |
| 5.1 <i>Screenshot</i> proyek Piktora pada <i>commit</i> 315d374 (31 Oktober 2016) - <i>commit</i> 89000be (12 Januari 2018). . . . . | 44 |
| 5.2 Salah satu <i>commit</i> yang terdapat pada <i>file</i> hasil animasi. Terdapat judul dibagian pojok kiri bawah. . . . .         | 45 |
| 5.3 Salah satu <i>commit</i> yang terdapat pada <i>file</i> hasil animasi. Terdapat logo dibagian pojok kanan bawah. . . . .         | 45 |
| 5.4 Salah satu <i>commit</i> pada <i>file</i> hasil animasi jika terdapat satu argumen <i>-capture-url</i> . . . . .                 | 49 |
| 5.5 Salah satu <i>commit</i> pada <i>file</i> hasil animasi jika terdapat dua argumen <i>-capture-url</i> . . . . .                  | 49 |
| 5.6 Salah satu <i>commit</i> pada <i>file</i> hasil animasi jika terdapat tiga argumen <i>-capture-url</i> . . . . .                 | 49 |
| 5.7 Salah satu <i>commit</i> pada <i>file</i> hasil animasi jika terdapat empat argumen <i>-capture-url</i> . . . . .                | 50 |
| 5.8 Tampilan pada Firefox <i>browser</i> saat dikontrol oleh FirefoxDriver. . . . .  | 51 |
| 5.9 Tampilan pada Opera <i>browser</i> saat dikontrol oleh OperaDriver. . . . .  | 52 |
| 5.10 Tampilan halaman <i>web</i> pada <i>browser</i> saat dikontrol oleh EdgeDriver. . . . .   | 53 |
| 5.11 Tampilan halaman <i>web</i> pada <i>browser</i> saat tidak dikontrol oleh EdgeDriver. . . . .                                   | 53 |
| 5.12 Keadaan awal dari <i>browser</i> saat dikontrol oleh InternetExplorerDriver. . . . .  | 54 |
| 5.13 Tampilan pada <i>browser</i> saat dikontrol oleh InternetExplorerDriver. . . . .  | 54 |
| 5.14 Sebagian hasil animasi dari situs <i>web</i> Netflix Open Source Software Center. . . . .                                       | 55 |
| 5.15 Sebagian hasil animasi dari situs <i>web</i> Bootstrap. . . . .   | 56 |
| 5.16 Sebagian hasil animasi dari situs <i>web</i> IBM Open Source. . . . .   | 57 |
| 5.17 Sebagian hasil animasi dari situs <i>web</i> React. . . . .   | 58 |
| 5.18 Sebagian hasil animasi dari situs <i>web</i> Yelp. . . . .  | 59 |



## **DAFTAR TABEL**

|     |   |    |
|-----|---|----|
| 3.1 | <i>Scenario case</i> membangkitkan animasi <i>timelapse</i> . . . . .               | 31 |
| 5.1 | Tabel pengujian fungsional . . . . .  | 46 |
| 5.2 | Tabel pengujian fungsional . . . . .  | 47 |
| 5.3 | Tabel pengujian fungsional . . . . .  | 48 |
| 5.4 | Tabel hasil pengujian eksperimental menggunakan beberapa situs <i>web</i> . . . . . | 59 |



1

## BAB 1

2

### PENDAHULUAN

#### 3 1.1 Latar Belakang

4 Git merupakan perangkat lunak *Version Control Systems*[1]. *Version Control* adalah sistem yang  
5 merekam perubahan pada *file* atau sekumpulan *file* dari waktu ke waktu. Perubahan yang terjadi  
6 pada repositori dicatat oleh Git dalam bentuk histori *commit*. Setiap *commit* mengandung informasi  
7 mengenai perubahan yang terjadi pada repositori, waktu perubahan, dan orang yang melakukan  
8 perubahan. *Database* pada *git* tidak bersifat terpusat, melainkan terdistribusi. Setiap orang yang  
9 terlibat mempunyai *database* lokal pada masing-masing komputer, sehingga pengelolaan perangkat  
10 lunak dapat dilakukan secara *online* dan *offline*.

11 JGit adalah *library* Java murni yang mengimplementasikan Git *version control systems*[2]. JGit  
12 dikembangkan oleh Eclipse Foundation. JGit bersifat *open source*. Dengan menggunakan JGit,  
13 operasi-operasi dalam Git dapat diakses melalui program Java.

14 Selenium adalah seperangkat alat yang secara khusus digunakan untuk mengotomatisasi *web*  
15 *browsers*[3]. Dengan menggunakan Selenium WebDriver, pengguna dapat memasukkan *script* bahasa  
16 pemrograman tertentu untuk melakukan pengujian. Bahasa pemrograman yang didukung yaitu C#,  
17 Java, Perl, PHP, Python, Ruby, dan JavaScript. Selenium WebDriver dapat melakukan pengujian  
18 pada Google Chrome *browser*, Firefox *browser*, Opera *browser*, Safari *browser*, Internet Explorer  
19 *browser*, dan Microsoft Edge *browser*.

20 Dengan adanya histori *commit* pada Git, perkembangan suatu proyek perangkat lunak dapat  
21 dipantau. *Programmer* bisa belajar dari proses perkembangan perangkat lunak pada *commit*  
22 sebelumnya dan bisa membuat perkembangan perangkat lunak menjadi lebih efisien. Akan tetapi,  
23 untuk perangkat lunak yang memiliki banyak *commit*, pemantauan progres dapat memakan waktu  
24 yang lama. Karena itu proses perkembangan perangkat lunak perlu divisualisasikan.

25 Pada skripsi ini, akan dibuat sebuah perangkat lunak yang dapat membangun animasi *timelapse*  
26 dari pengembangan proyek perangkat lunak berbasis *web*. Yang akan dibuat animasinya adalah  
27 halaman-halaman *web* dari suatu perangkat lunak berbasis *web* yang terkam oleh Git. Perangkat  
28 lunak ini dibangun menggunakan bahasa Java. Perangkat lunak ini menggunakan antarmuka  
29 terminal/konsol. Masukan perangkat lunak diambil dari argumen Command Line. Perangkat lunak  
30 dibuat menggunakan bantuan *library* JGit, dan Selenium WebDriver. *Output* yang dihasilkan dari  
31 program ini adalah *file* hasil animasi yang bertipe GIF.

32 **1.2 Rumusan Masalah**

33 Rumusan masalah dari skripsi ini adalah sebagai berikut:

- 34 1. Bagaimana cara membangkitkan animasi *timelapse* pada pengembangan proyek perangkat  
35 lunak berbasis web?
- 36 2. Bagaimana cara mengimplementasikan aplikasi untuk membangkitkan *timelapse* pada pe-  
37 ngembangan proyek perangkat lunak berbasis web?
- 38 3. Kesimpulan apa yang dapat diambil dari animasi *timelapse* pada beberapa situs *web* Open  
39 Source?

40 **1.3 Tujuan**

41 Tujuan dari skripsi ini adalah sebagai berikut:

- 42 1. Mengetahui cara untuk membangkitkan animasi *timelapse* pada pengembangan proyek per-  
43 angkat lunak berbasis web.
- 44 2. Mengetahui cara untuk mengimplementasikan aplikasi untuk membangkitkan *timelapse* pada  
45 pengembangan proyek perangkat lunak berbasis web.
- 46 3. Mendapatkan kesimpulan dari animasi *timelapse* pada beberapa situs *web* Open Source.

47 **1.4 Batasan Masalah**

48 Batasan masalah pada skripsi ini adalah sebagai berikut:

- 49 1. Perangkat lunak ini hanya membangkitkan animasi *timelapse* untuk perangkat lunak berbasis  
50 *web* yang repositorinya terekam oleh Git.
- 51 2. Jumlah halaman *web* pada konfigurasi maksimal berjumlah empat halaman.
- 52 3. *Setup* situs *web* yang akan dibuat animasinya harus bisa melalui *terminal* (tanpa GUI).

53 **1.5 Metodologi**

54 Metodologi penelitian yang digunakan dalam skripsi ini adalah sebagai berikut:

- 55 1. Melakukan studi literatur tentang Git, JGit, Selenium WebDriver, dan Apache Commons  
56 CLI.
- 57 2. Melakukan analisis penggunaan Selenium WebDriver dan JGit untuk membangkitkan animasi  
58 *timelapse*.
- 59 3. Merancang perangkat lunak.
- 60 4. Membangun perangkat lunak.
- 61 5. Melakukan eksperimen dan pengujian pada perangkat lunak.

## 62    1.6    Sistematika Pembahasan

- 63    Setiap bab dalam penelitian ini memiliki sistematika penulisan yang dijelaskan ke dalam poin-poin  
64    sebagai berikut:
- 65    1. Bab 1: Pendahuluan, yaitu membahas mengenai gambaran umum penelitian ini. Berisi  
66    tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metode penelitian, dan  
67    sistematika penulisan.
- 68    2. Bab 2: Dasar Teori, yaitu membahas mengenai teori-teori yang mendukung berjalannya  
69    penelitian ini. Berisi tentang teori Git, JGit, Selenium WebDriver, dan Apache Commons  
70    CLI.
- 71    3. Bab 3: Analisis, yaitu membahas mengenai analisa masalah. Berisi tentang analisis aplikasi  
72    sejenis, analisis penggunaan JGit dan Selenium WebDriver untuk membangkitkan animasi  
73    timelapse, analisis fitur aplikasi yang dibangun, dan prapengujian.
- 74    4. Bab 4: Perancangan, yaitu membahas mengenai perancangan yang dilakukan sebelum melakuk  
75    ukan tahapan implementasi. Berisi tentang perancangan kelas dan perancangan antarmuka  
76    dari perangkat lunak.
- 77    5. Bab 5: Implementasi dan Pengujian, yaitu membahas mengenai implementasi dan pengujian  
78    aplikasi yang telah dilakukan. Berisi tentang implementasi dan hasil pengujian perangkat  
79    lunak.
- 80    6. Bab 6: Kesimpulan dan Saran, yaitu membahas hasil kesimpulan dari keseluruhan penelitian  
81    ini dan saran-saran yang dapat diberikan untuk penelitian berikutnya.



## BAB 2

### LANDASAN TEORI

<sup>84</sup> Pada bab ini dibahas dasar teori yang mendukung berjalannya skripsi ini. Dasar teori yang dibahas  
<sup>85</sup> yaitu Git, JGit, Selenium WebDriver, dan Apache Commons CLI.

<sup>86</sup> **2.1 Git**

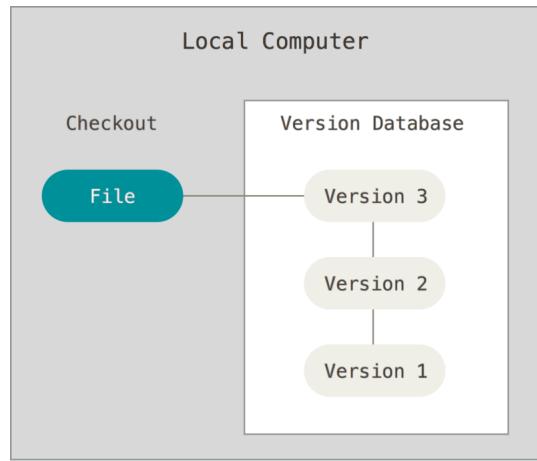
<sup>87</sup> Git merupakan perangkat lunak *Version Control Systems*. Pada subbab ini, dijelaskan mengenai  
<sup>88</sup> *Version Control Systems*, cara kerja Git, Git *checkout*, dan operasi-operasi dasar pada Git. Subbab  
<sup>89</sup> ini mengacu pada [1].

<sup>90</sup> **2.1.1 Version Control Systems**

<sup>91</sup> *Version Control Systems* adalah sistem yang merekam perubahan pada *file* atau sekumpulan *file*  
<sup>92</sup> dari waktu ke waktu. *Version Control Systems* biasanya digunakan untuk merekam file yang berisi  
<sup>93</sup> *source code program*, tetapi pada kenyataannya *Version Control Systems* dapat merekam hampir  
<sup>94</sup> semua jenis file dalam komputer. Terdapat tiga jenis *Version Control Systems*, yaitu: *Local Version*  
<sup>95</sup> *Control Systems*, *Centralized Version Control Systems*, dan *Distributed Version Control Systems*.

<sup>96</sup> **Local Version Control Systems**

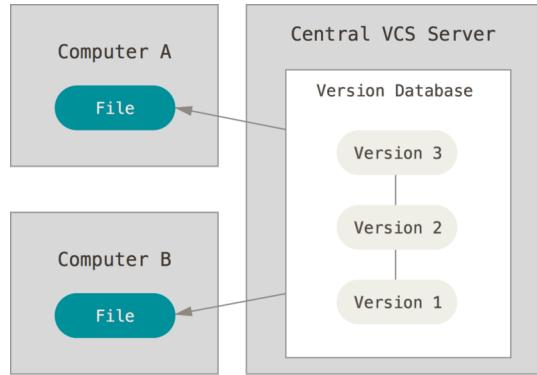
<sup>97</sup> Metode *version-controlled* yang banyak digunakan orang adalah dengan cara menyalin sekumpulan  
<sup>98</sup> *file* ke direktori lain. Namun cara tersebut rentan terhadap *error*. Misalnya, terdapat direktori A  
<sup>99</sup> dan B, pengguna ingin mengubah *file* yang terdapat pada direktori B, tetapi pengguna lupa kalau  
<sup>100</sup> dia sedang berada di direktori A, maka pengguna mengubah *file* pada direktori yang salah. Untuk  
<sup>101</sup> mengatasi masalah tersebut, *programmer* mengembangkan *Local Version Control Systems*.



Gambar 2.1: Local version control[1].

102 Gambar 2.1 merupakan struktur dari *Local Version Control Systems*. *Database local Version*  
 103 *Control Systems* ini tersimpan pada *local directory* di komputer. *Database* ini menyimpan perubahan  
 104 *file* ke dalam beberapa versi atau *state*.

## 105 Centralized Version Control Systems



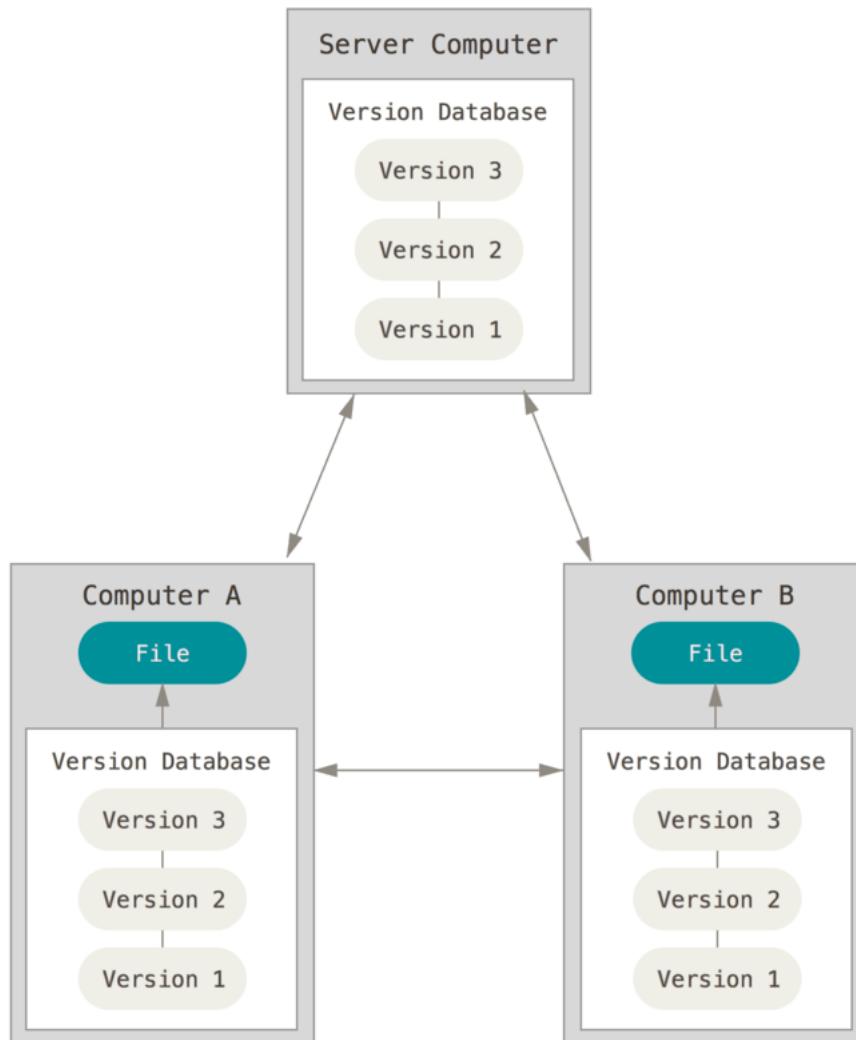
Gambar 2.2: Centralized version control[1].

106 *Local Version Control* hanya menyimpan *file* pada satu komputer saja. Muncul masalah baru ketika  
 107 *user* ingin berkolaborasi dengan *user* lain. Untuk mengatasi masalah ini dikembangkan *Centralized*  
 108 *version control*. Gambar 2.2 merupakan struktur dari *Centralized Version Control Systems*. Dalam  
 109 *Centralized Control Version Systems* terdapat sebuah *server* yang menyimpan setiap versi *file*, dan  
 110 *klien* yang dapat melakukan *checkout file*.

111 Sistem *Centralized Version Control Systems* memiliki beberapa kelebihan. Setiap *user* dapat  
 112 mengetahui pekerjaan yang dilakukan oleh *user* lain. Administrator dapat lebih mudah mengontrol  
 113 *database Centralized Version Control Systems* dibandingkan dengan *database Local Version Control*  
 114 *Systems* dari setiap klien.

115 Sistem *Centralized Version Control Systems* memiliki kelemahan. Jika *server* pusat *Centralized*  
 116 *Version Control Systems* mati, maka perubahan pada *file* tidak bisa disimpan. Klien juga tidak  
 117 dapat melakukan kolaborasi dengan klien lain. Jika *harddisk* pada server rusak, maka semua versi  
 118 *file* akan hilang.

<sup>119</sup> **Distributed Version Control Systems**

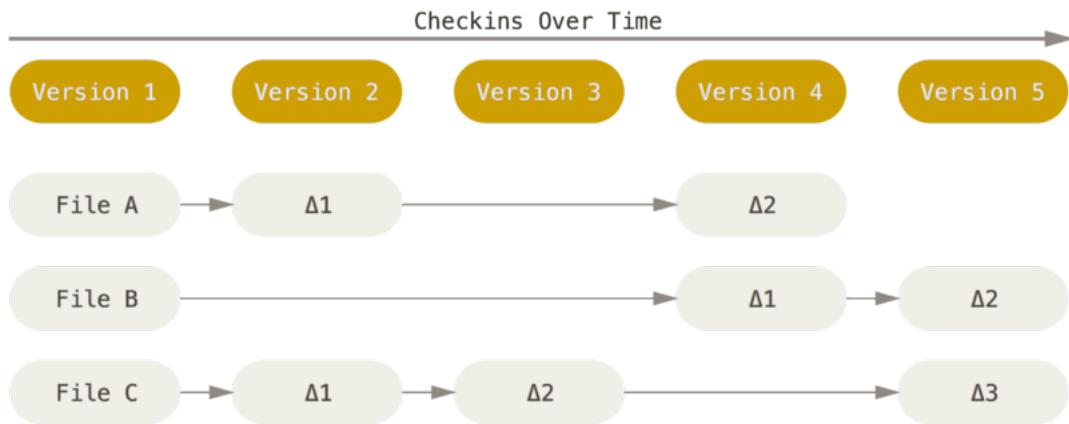


Gambar 2.3: Distributed version control<sup>[1]</sup>.

<sup>120</sup> Gambar 2.3 merupakan struktur dari *Distributed Version Control Systems*. Dalam sebuah DVCS (seperti Git, Mercurial, Bazaar atau Darcs), klien tidak hanya melakukan *checkout* untuk *snapshot* terakhir setiap *file*, namun klien juga memiliki salinan dari repositori tersebut. Dengan kata lain setiap klien memiliki *version database local* pada komputernya. Jika server pusat mati, klien masih bisa melakukan kolaborasi dan klien manapun dapat mengirimkan kembali salinan repositori ke *server*.

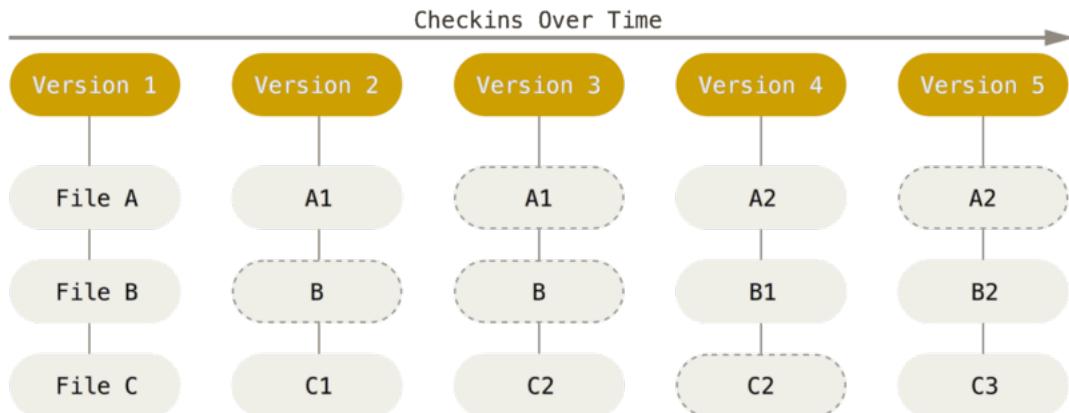
<sup>126</sup> **2.1.2 Cara Kerja Git**

<sup>127</sup> Salah satu perbedaan antara Git dengan VCS lainnya adalah dalam cara Git memperlakukan datanya. Kebanyakan sistem *Version Control Systems* lain menyimpan informasi sebagai daftar perubahan *file*. Pada Gambar 2.4, terdapat tiga *file*. *Version Control Systems* menyimpan *file* A, B, dan C pada versi pertama saja. Untuk versi kedua dan seterusnya yang disimpan adalah perubahan pada setiap *file*. Sistem ini disebut juga sebagai *delta-based Version Control Systems*.



Gambar 2.4: Menyimpan data sebagai *snapshots* dari *project*[1].

132 Berbeda dengan *Version Control Systems* lainnya, Git memperlakukan datanya sebagai sebuah  
 133 kumpulan *snapshot* dari sebuah miniatur *file system*. Setiap kali dilakukan *commit*, git merekam  
 134 *state* dari sekumpulan *file* dan menyimpannya sebagai *snapshot*. Gambar 2.5, menunjukkan *snapshots*  
 135 dari *file* A, B, dan C. Pada versi kedua, *file* B tidak mengalami perubahan, sehingga yang disimpan  
 136 adalah *reference* dari *file* B pada versi sebelumnya.



Gambar 2.5: Menyimpan data sebagai perubahan terhadap versi dasar dari setiap *file*[1].

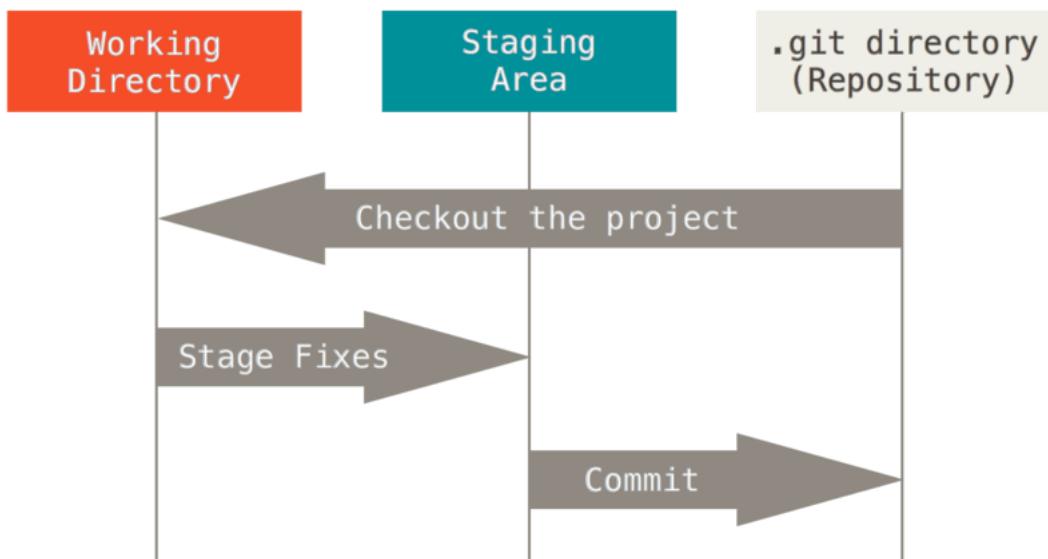
### 137 State pada Git

138 Terdapat tiga *state* pada Git yaitu *committed*, *modified*, and *staged*. *Committed* adalah *state* dimana  
 139 data sudah disimpan di *local database*. *Modified* adalah *state* dimana terdapat perubahan pada *file*,  
 140 namun *file* tersebut belum di *commit* ke *database*. *Staged* adalah *state* dimana *file* telah ditandai  
 141 untuk kemudian dilakukan *commit*.

142 Terdapat tiga bagian utama dari sebuah *project* Git yaitu direktori Git, *working directory*, dan  
 143 *staging area*. Direktori Git merupakan tempat dimana Git menyimpan *metadata* dan *object database*  
 144 dari *project*. *Working tree* adalah suatu *snapshot* dari *project*. Sekumpulan *file* ini diambil dari  
 145 *database* di direktori Git dan ditempatkan pada *disk* untuk digunakan dan dimodifikasi. *Staging*  
 146 area adalah suatu *file*, dimana *file* ini menyimpan daftar *file* yang telah ditandai untuk kemudian  
 147 dilakukan *commit*. *File staging area* terdapat pada direktori Git. Untuk lebih jelasnya, lihat Gambar  
 148 2.6.

<sup>149</sup> Alur kerja dari Git adalah sebagai berikut:

- <sup>150</sup> 1. Melakukan modifikasi pada *file*.
- <sup>151</sup> 2. Menandai perubahan pada *file* dan memindahkannya ke *staging area*.
- <sup>152</sup> 3. Mengambil *file* dari *staging area* dan menyimpan *snapshot* ke direktori Git. Proses ini disebut dengan *commit*.



Gambar 2.6: *Working tree*, *Staging area*, dan Git direktori[1].

## <sup>154</sup> Commit

<sup>155</sup> Commit merupakan sebuah *snapshot* dari suatu *file* atau direktori. Dengan kata lain yang disimpan oleh Git adalah *state* dari *file* atau direktori pada suatu waktu. *File* disimpan oleh Git dalam bentuk *blob*, sedangkan direktori disimpan dalam bentuk *tree*. *Commit*, *blob*, dan *tree* merupakan Git Object. Setiap Git Object memiliki ID berupa SHA-1 *hash value* dengan panjang empat puluh karakter heksadesimal.

Listing 2.1: Contoh histori commit dalam pengembangan perangkat lunak

```

16b commit 89000be7ce7d16f006813cddefb4ec6d70d15ed6 (HEAD -> master, origin/master,
161           origin/HEAD)
162 Author: Hizkia Steven <xvii.hs@gmail.com>
163 Date:   Fri Jan 12 12:25:30 2018 +0700
164
165     Update new company address
166
167 commit 6a085c1c37949e6308cfe06a117302e528388e54
168 Author: Hizkia Steven <xvii.hs@gmail.com>
169 Date:   Tue Dec 12 14:38:38 2017 +0700
170
171     Update company address
172
  
```

```

13 commit 9f041ef239bfe236ab4d679ad698d773a8ba6f56
14 Author: TommyAdhityaThe <toms.warior@gmail.com>
15 Date: Mon May 15 10:40:16 2017 +0700
16
17     set insta url to https://www.instagram.com/piktorastudio/
18
19 commit 38711f0cc8f487aac62babac10c1185f5ee14d33
20 Author: Tommy Adhitya The <toms.warior@gmail.com>
21 Date: Mon Apr 17 15:15:03 2017 +0700
22
23     fix bug ugly display when projects too high

```

Listing 2.1 menunjukkan sebagian histori *commit* pada *branch master*. Baris pertama menunjukkan *commit ID* berupa *SHA-1 hash value*, dengan panjang empat puluh karakter heksadesimal. Baris kedua menunjukkan orang yang melakukan *commit* dan alamat emailnya. Baris ketiga menunjukkan waktu *commit*. Baris terakhir berisi deskripsi dari *commit* tersebut.

HEAD dan *master* pada Listing 2.1, disebut juga dengan Git Ref / Reference. Ref merupakan suatu variabel yang menyimpan ID (SHA-1 hash value) dari suatu Git Object. Pada Listing 2.1, Ref *master* menyimpan ID dari *commit* yang isinya adalah 89000be7ce7d16f006813cddefb4ec6d70d15ed6. Ref HEAD tidak secara langsung menyimpan ID dari *commit*, melainkan *pointer* ke Ref *master*. Ref yang menyimpan *pointer* ke Ref lain disebut juga dengan *symbolic Ref*.

Ref origin/master merupakan Ref *master* yang berada pada *remote repository*. Ref origin/HEAD merupakan ref HEAD yang berada pada *remote repository*. *Remote repository* merupakan repositori yang disimpan pada suatu *server*, misalnya *server GitHub*.

### 2.1.3 Operasi Dasar pada Git

Pada subbab ini dijelaskan mengenai operasi dasar dalam Git dan sintaks-sintaksnya. Sintaks-sintaksnya ini dimasukkan pada Git *command line*. Berikut ini adalah operasi-operasi dasar dalam Git:

#### 1. Init

Operasi ini digunakan untuk membuat repositori lokal baru dengan nama tertentu. Bisa juga digunakan untuk merekam direktori yang sudah ada. Berikut adalah sintaks untuk melakukan operasi *init*:

```
$ git init [project-name]
```

#### 2. Add

Operasi ini digunakan untuk menandai perubahan pada *file* dan memindahkan *file* tersebut ke *staging area*. Operasi ini juga digunakan untuk menambahkan *file* yang dipantau perubahannya. Berikut adalah sintaks untuk melakukan operasi add:

```
$ git add [file]
```

#### 3. Commit

Operasi ini digunakan untuk merekam *snapshot* atau *state file* atau sekumpulan *file*. Operasi ini juga digunakan untuk memindahkan *file* yang berada di *staging area* ke direktori Git. Berikut adalah sintaks untuk melakukan operasi *commit*:

```
$ git commit [-m <descriptive message>]
```

## 215 4. Branch

216 Operasi ini digunakan untuk menampilkan semua *branch* baik itu yang terdapat di *local*  
217 *repository* maupun *remote repository*, menampilkan *branch* yang terdapat pada *remote reposi-*  
218 *tory*, membuat *branch* baru, dan menghapus *branch*. Berikut adalah sintaks-sintaks untuk  
219 melakukan operasi *branch*:

```
220 $ git branch [-a]  
221 $ git branch [-r]  
222 $ git branch [branch-name]  
223 $ git branch [-d <branch-name>]  
224 $ git branch [-D <branch-name>]
```

## 225 5. Diff

226 Operasi ini digunakan untuk menampilkan perbedaan pada *file* yang belum masuk *staging*  
227 *area*, dan menampilkan perbedaan pada *file* yang berada di *staging area* dengan *file* di *commit*  
228 sebelumnya. Berikut adalah sintaks-sintaks untuk melakukan operasi *diff*:

```
229 $ git diff  
230 $ git diff [-staged]
```

## 232 6. Clone

233 Operasi ini digunakan untuk menyalin repositori Git yang berada di komputer lain atau suatu  
234 *server*. Berikut adalah sintaks untuk melakukan operasi *clone*:

```
235 $ git clone <url>
```

## 236 7. Fetch

237 Operasi ini digunakan untuk mengambil data dari *remote* repositori ke repositori lokal. Berikut  
238 adalah sintaks untuk melakukan operasi *fetch*:

```
239 $ git fetch [bookmark]
```

## 240 8. Merge

241 Operasi ini digunakan untuk menggabungkan *branch* tertentu dengan *branch* yang sedang  
242 aktif. Operasi ini juga digunakan untuk menggabungkan data yang diambil dari *remote*  
243 repositori dengan data pada *working directory*. Berikut adalah sintaks untuk melakukan  
244 operasi *merge*:

```
245 $ git merge [branch]/[bookmark]
```

## 246 9. Pull

247 Operasi ini adalah gabungan dari operasi *fetch* dan *merge*. Berikut adalah sintaks untuk  
248 melakukan operasi *pull*:

```
249 $ git pull
```

## 250 10. Push

251 Operasi ini digunakan untuk mengirim data pada reposipori Git lokal ke *remote repository*.  
252 Berikut adalah sintaks untuk melakukan operasi *push*:

```
253 $ git push [bookmark] [branch]
```

254 11. Checkout

255 Operasi ini digunakan untuk berpindah ke *branch* atau *commit* tertentu, setelah itu mem-  
256 perbarui *file* pada *working directory* berdasarkan *branch* atau *commit* tersebut. Berikut ini  
257 adalah sintaks-sintaks untuk operasi *checkout*:

258 \$ git checkout [commit ID]  
259 \$ git checkout [branch-name]

260 12. Log

261 Operasi ini digunakan untuk menampilkan semua histori *commit* pada *branch* yang sedang  
262 aktif. Berikut ini adalah sintaks untuk melakukan operasi *log*:

263 \$ git log

264 13. Reset

265 Operasi ini digunakan untuk memindahkan posisi *HEAD* ke *commit* tertentu, selain itu secara  
266 opsional melakukan *reset* pada *staging area* dan *working tree* berdasarkan tipe *reset*. Terdapat  
267 tiga tipe reset yaitu:

268 • Soft

269 Pada tipe ini, tidak dilakukan *reset* pada *staging area* dan *working directory*.

270 • Hard

271 Pada tipe ini, dilakukan *reset* pada *staging area* dan *working tree* sehingga perubahan  
272 yang terdapat pada *staging area* dan *working directory* akan hilang.

273 • Mixed

274 Pada tipe ini, dilakukan *reset* pada *staging area* sehingga perubahan yang terdapat pada  
275 *staging area* akan dipindahkan ke *working directory*.

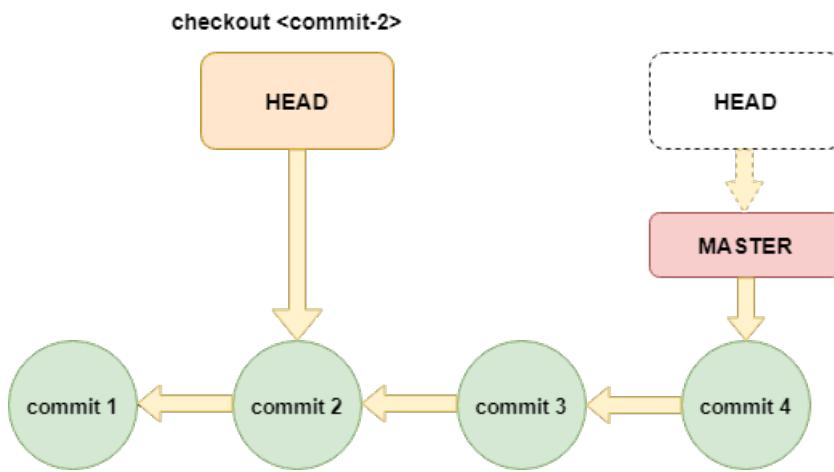
276 Berikut ini adalah sintaks untuk melakukan operasi *reset*:

277 \$ git reset -hard [commit]  
278 \$ git reset -mixed [commit]  
279 \$ git reset -soft [commit]

280 2.1.4 Git Checkout

281 Seperti yang sudah dijelaskan pada subbab 2.1.3, *checkout* dapat digunakan untuk berpindah  
282 ke *branch* atau *commit* tertentu. Operasi *checkout* dapat dilakukan menggunakan sintaks \$ git  
283 *checkout* diikuti dengan nama *branch* atau *commit ID*. Gambar 2.7 menunjukkan contoh *checkout*  
284 pada *commit*. Posisi awal *HEAD* menunjuk pada *branch master*, setelah dilakukan *checkout* ke  
285 *commit 2*, posisi *HEAD* menunjuk pada *commit 2*. *Working directory* diperbarui berdasarkan *state*  
286 pada *commit 2*.

287 *HEAD* yang menunjuk langsung ke suatu *commit* disebut dengan *detached HEAD*. Perubahan  
288 yang terjadi pada *detached HEAD* tidak akan terekam oleh Git. Jika terdapat perubahan, kemudian  
289 dilakukan *checkout commit* atau *branch*, perubahan tersebut akan hilang. *HEAD* dapat dipindahkan  
290 ke posisi semula(menunjuk pada *branch master*) dengan menggunakan sintaks \$ git *checkout*  
291 *master*.



Gambar 2.7: *Checkout* pada *commit*

## 292 2.2 JGit

293 JGit adalah *library* Java murni yang mengimplementasikan Git *version control systems*[2]. Dengan  
294 menggunakan JGit, operasi-operasi dalam Git bisa dilakukan melalui program Java. Pada subbab  
295 berikut dijelaskan beberapa kelas dari *library* JGit. Subbab ini mengacu pada [4].

### 296 2.2.1 Kelas Repository

297 Kelas ini merepresentasikan repositori Git. Berikut ini adalah beberapa *method* dalam kelas ini:

- 298 • public void create()  
299 Berfungsi untuk membuat repositori Git baru.
- 300 • public String getBranch()  
301 Berfungsi untuk mendapatkan nama *branch* yang ditunjuk oleh *HEAD*.  
302 Kembalian: nama dari *branch* yang sedang aktif, contohnya *master*.
- 303 • public Ref getRef(String name)  
304 Berfungsi untuk mendapatkan *reference* berdasarkan nama yang diberikan.  
305 Parameter: nama dari *reference*.  
306 Kembalian: *object* bertipe Ref.

### 307 2.2.2 Kelas Ref

308 Ref di dalam Git merupakan sebuah variabel yang menyimpan ID dari Git Object. Berikut adalah  
309 beberapa *method* yang dimiliki oleh kelas ini:

- 310 • String getName()  
311 Berfungsi untuk mengembalikan nama dari suatu Ref.  
312 Kembalian: nama dari Ref.
- 313 • ObjectId getObjectId()  
314 Berfungsi untuk mengembalikan ID dari suatu Git Object.

315 Kembalian: *object* bertipe ObjectId. Di dalam kelas ObjectId terdapat *method* getName()  
316 yang berfungsi untuk mengembalikan ID dari Git Object berupa SHA-1 *hash value*.

317 **2.2.3 Kelas FileRepository**

318 Kelas ini merupakan turunan dari kelas *Repository*. Berikut ini adalah *constructor* dari kelas ini:

- 319 • public FileRepository(String gitDir) throws IOException  
320 *Constructor* ini membuat representasi dari repositori Git. *Constructor* ini melempar IOException jika repositori tidak bisa diakses.  
321 Parameter: *path* dari suatu repositori Git.

323 **2.2.4 Kelas Git**

324 Kelas ini menyediakan API yang mirip Git Command Line untuk berinteraksi dengan repositori  
325 Git. Berikut ini adalah *constructor* dan beberapa *method* dalam kelas ini:

- 326 • public Git(Repository repo)  
327 *Constructor* ini membuat objek Git yang digunakan untuk berinteraksi dengan repositori Git.  
328 Parameter: objek *Repository* yang digunakan untuk berinteraksi. Parameter tidak boleh  
329 bernilai *null*.
- 330 • public static InitCommand init()  
331 *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *init*.  
332 Kembalian: objek *InitCommand* yang berfungsi untuk mengumpulkan parameter opsional  
333 dan akhirnya mengeksekusi operasi *init*.
- 334 • public AddCommand add()  
335 *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *add*.  
336 Kembalian: objek *AddCommand* yang berfungsi untuk mengumpulkan parameter opsional  
337 dan akhirnya mengeksekusi operasi *add*.
- 338 • public LogCommand log()  
339 *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *log*.  
340 Kembalian: objek *LogCommand* yang berfungsi untuk mengumpulkan parameter opsional  
341 dan akhirnya mengeksekusi operasi *log*.
- 342 • public ListBranchCommand branchList()  
343 *Method* ini mengembalikan objek *command* yang digunakan untuk mengeksekusi operasi  
344 *branch*. Operasi ini setara dengan sintaks *git branch -r/-a*.  
345 Kembalian: objek dengan tipe *ListBranchCommand*.
- 346 • public CheckoutCommand checkout()  
347 *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *checkout*.  
348 Kembalian: objek *ChecoutCommand* yang berfungsi untuk mengumpulkan parameter opsional  
349 dan akhirnya mengeksekusi operasi *checkout*.

- 350     ● public CommitCommand commit()  
351         *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *commit*.  
352         Kembalian: objek *CommitCommand* yang berfungsi untuk mengumpulkan parameter opsional  
353         dan akhirnya mengeksekusi operasi *commit*.
  
- 354     ● public FetchCommand fetch()  
355         *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *fetch*.  
356         Kembalian: objek *FetchCommand* yang berfungsi untuk mengumpulkan parameter opsional  
357         dan akhirnya mengeksekusi operasi *fetch*.
  
- 358     ● public PushCommand push()  
359         *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *push*.  
360         Kembalian: objek *PushCommand* yang berfungsi untuk mengumpulkan parameter opsional  
361         dan akhirnya mengeksekusi operasi *push*.
  
- 362     ● public DiffCommand diff()  
363         *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *diff*.  
364         Kembalian: objek *DiffCommand* yang berfungsi untuk mengumpulkan parameter opsional  
365         dan akhirnya mengeksekusi operasi *diff*.
  
- 366     ● public static CloneCommand cloneRepository()  
367         *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *clone*.  
368         Kembalian: objek *CloneCommand* yang berfungsi untuk mengumpulkan parameter opsional  
369         dan akhirnya mengeksekusi operasi *clone*.
  
- 370     ● public MergeCommand merge()  
371         *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *merge*.  
372         Kembalian: objek *MergeCommand* yang berfungsi untuk mengumpulkan parameter opsional  
373         dan akhirnya mengeksekusi operasi *merge*.
  
- 374     ● public PullCommand pull()  
375         *Method* ini mengembalikan objek *command* untuk mengeksekusi operasi *pull*.  
376         Kembalian: objek *PullCommand*.
  
- 377     ● public CreateBranchCommand branchCreate()  
378         *Method* ini mengembalikan objek *command* untuk membuat *branch* baru.  
379         Kembalian: objek *CreateBranchCommand*.
  
- 380     ● public ListBranchCommand branchList()  
381         *Method* ini mengembalikan objek *command* untuk menampilkan daftar *branch*.  
382         Kembalian: objek *ListBranchCommand*.
  
- 383     ● public DeleteBranchCommand branchDelete()  
384         *Method* ini mengembalikan objek *command* untuk menghapus *branch*.  
385         Kembalian: objek *DeleteBranchCommand*.

- 386 • public ResetCommand reset()  
387     *Method ini mengembalikan objek *command* untuk mengeksekusi operasi *reset*.*  
388     *Kembalian: objek *ResetCommand*.*

389 **2.2.5 Kelas CheckoutCommand**

390 Kelas yang digunakan untuk melakukan operasi *checkout*. Berikut adalah beberapa *method* yang  
391 terdapat pada kelas ini:

- 392 • public CheckoutCommand setName(String name)  
393     Menentukan nama *branch* atau *commit* untuk melakukan checkout.  
394     Parameter: nama dari *branch* atau *commit*.  
395     Kembalian: *object CheckoutCommand*.  
396 • public Ref call()  
397     Berfungsi untuk menjalankan operasi *checkout*.

398 **2.2.6 Kelas ListBranchCommand**

399 Kelas yang digunakan untuk melakukan operasi *branch* dan mendapatkan daftar *branch*. Berikut  
400 adalah beberapa *method* yang terdapat pada kelas ini:

- 401 • public ListBranchCommand setListMode(ListBranchCommand.ListMode listMode)  
402     Menentukan mode yang digunakan untuk mendapatkan *branch*. Jika parameter tidak dima-  
403     sukkan, hanya daftar *branch* lokal yang didapatkan.  
404     Parameter: mode yang digunakan untuk mendapatkan *branch* berupa konstanta.  
405     Kembalian: *object CheckoutCommand*.  
406 • public List<Ref> call()  
407     Berfungsi untuk menjalankan operasi *branch*.  
408     Kembalian: daftar *branch* dalam bentuk List<Ref>.

409 **2.2.7 Enum ListBranchCommand.ListMode**

410 Merupakan sebuah Enum yang berisi mode yang digunakan untuk mendapatkan *branch*. Berikut  
411 ini adalah konstanta yang terdapat pada Enum ini:

- 412 • public static final ListBranchCommand.ListMode ALL  
413     Mode ini digunakan untuk mendapatkan semua daftar *branch* baik lokal maupun *remote*.  
414     Mode ini setara dengan *option -a* pada sintaks git branch -a.  
415 • public static final ListBranchCommand.ListMode REMOTE  
416     Mode ini digunakan untuk mendapatkan daftar *remote branch*. Mode ini setara dengan *option*  
417     -r pada sintaks git branch -r.

418 **2.2.8 Kelas LogCommand**

419 Kelas yang digunakan untuk melakukan operasi *log*. Berikut adalah *method* yang terdapat pada  
420 kelas ini:

- 421 • public Iterable<RevCommit> call()

422 Berfungsi untuk menjalankan operasi *log*.

423 Kembalian: histori *commit* pada *branch* yang sedang aktif.

424 **2.2.9 Kelas ResetCommand**

425 Kelas yang digunakan untuk melakukan operasi *reset*. Berikut adalah beberapa *method* yang  
426 terdapat pada kelas ini:

- 427 • public ResetCommand setMode(ResetCommand.ResetType mode)

428 Menentukan tipe *reset*.

429 Parameter: tipe *reset*.

430 Kembalian: *object* ResetCommand.

- 431 • public ResetCommand setRef(String ref)

432 Mengatur nama *reference* tujuan dalam operasi *reset*. Secara default, *reference* yang digunakan  
433 adalah *HEAD*.

434 Parameter: nama dari *reference*.

435 Kembalian: *object* ResetCommand.

- 436 • public Ref call()

437 Berfungsi untuk menjalankan operasi *reset*.

439 **2.2.10 Kelas ResetCommand.ResetType**

440 Merupakan *enumeration* yang menentukan tipe *reset* yang digunakan pada operasi *reset*. *Enumeration*  
441 tersebut adalah sebagai berikut:

- 442 • public static final ResetCommand.ResetType SOFT

443 Hanya mengubah posisi *HEAD*.

- 444 • public static final ResetCommand.ResetType HARD

445 Mengubah posisi *HEAD* ke *reference* tujuan. Selain itu, mengubah *staging area* dan *working*  
446 *directory* berdasarkan *reference* tersebut.

- 447 • public static final ResetCommand.ResetType MIXED

448 Mengubah posisi *HEAD* ke *reference* tujuan dan mengubah *staging area* berdasarkan *reference*  
449 tersebut.

450 **2.2.11 Kelas RevCommit**

451 Kelas ini merupakan *reference* ke *commit* yang ada di *Directed Acyclic Graph*. Berikut ini adalah  
452 beberapa *method* dari kelas ini:

- 453 • public final String getFullMessage()  
454 Berfungsi untuk mendapatkan deskripsi dari suatu *commit*.  
455 Kembalian: deskripsi dari suatu *commit* dalam bentuk String.
- 456 • public final String getShortMessage()  
457 Berfungsi untuk mendapatkan deskripsi dari suatu *commit*. Jika panjang deskripsi *commit*  
458 lebih dari satu baris, maka hanya baris pertama yang dikembalikan.  
459 Kembalian: baris pertama dari deskripsi *commit* dalam bentuk String.
- 460 • public final String getName()  
461 *Method* ini mengembalikan *commit ID*.  
462 Kembalian: *commit ID* berupa SHA-1 hash value dalam bentuk String.
- 463 • public final PersonIdent getAuthorIdent()  
464 Berfungsi untuk mendapatkan informasi mengenai *author* yang melakukan *commit*.  
465 Kembalian: objek *PersonIdent* yang memuat informasi tentang *author*(nama dan *email*) dan  
466 waktu dilakukannya *commit*.

### 467 2.2.12 Kelas PersonIdent

468 Kelas ini memberikan informasi mengenai *author* dari suatu *commit*. Berikut ini adalah beberapa  
469 *method* dari kelas ini:

- 470 • public String getName()  
471 Berfungsi untuk mengembalikan nama dari *author* yang melakukan *commit*.  
472 Kembalian: nama dari *author*.
- 473 • public String getEmailAddress()  
474 Berfungsi untuk mengembalikan alamat *email* dari *author* yang melakukan *commit*.  
475 Kembalian: alamat *email* dari *author*.
- 476 • public Date getWhen()  
477 Berfungsi mengembalikan waktu dilakukannya suatu *commit* oleh *author*.  
478 Kembalian: waktu dilakukannya suatu *commit* berupa objek Date.

## 479 2.3 Selenium WebDriver

480 Selenium adalah seperangkat alat yang secara khusus digunakan untuk mengotomatisasi *web*  
481 *browsers*[3]. Selenium mendukung bahasa pemrograman C#, Java, Perl, PHP, Python, Ruby,  
482 dan JavaScript. Selenium terdiri dari beberapa bagian, yaitu Selenium 1(Selenium RC), Selenium  
483 2(Selenium WebDriver), dan Selenium IDE. Selenium RC merupakan proyek utama Selenium untuk  
484 waktu yang lama, sebelum akhirnya bergabung dengan WebDriver menjadi Selenium 2. Selenium  
485 RC melakukan *automation test* dengan cara menginjeksi kode JavaScript ke *browser*. Selenium RC  
486 sekarang sudah *deprecated* dan tidak digunakan lagi. Selenium Webdriver merupakan gabungan  
487 dari Selenium RC dan WebDriver. Selenium IDE merupakan *extension* dari browser Chrome dan  
488 Firefox yang digunakan untuk melakukan *record and playback test* pada *browser*.

489 WebDriver merupakan kakas untuk mengotomatisasi pengujian pada aplikasi *web*[5]. WebDriver  
490 dapat berkomunikasi secara langsung dengan *browser* menggunakan *native support* pada *browser*  
491 tanpa melakukan injeksi kode JavaScript. WebDriver yang terdapat pada Selenium WebDriver antara  
492 lain ChromeDriver, FirefoxDriver, OperaDriver, SafariDriver, InternetExplorerDriver, EdgeDriver,  
493 dan HtmlUnitDriver.

494 ChromeDriver merupakan implementasi WebDriver yang mengontrol Chrome *browser*. Fire-  
495 foxDriver merupakan implementasi WebDriver yang mengontrol Firefox *browser*. OperaDriver  
496 merupakan implementasi WebDriver yang mengontrol Opera *browser*. SafariDriver merupakan  
497 implementasi WebDriver yang mengontrol Safari *browser*. InternetExplorerDriver merupakan  
498 implementasi WebDriver yang mengontrol Internet Explorer *browser*. EdgeDriver merupakan  
499 implementasi WebDriver yang mengontrol Microsoft Edge *browser*. HtmlUnitDriver merupakan im-  
500 plementasi WebDriver yang mengontrol HTMLUnit *browser* yang bersifat *headless*(tidak mempunyai  
501 GUI).

502 Pada skripsi ini *tools* Selenium yang digunakan hanya Selenium WebDriver. Bahasa pemro-  
503 graman yang digunakan adalah java. WebDriver yang digunakan adalah ChromeDriver. Pemilihan  
504 WebDriver ini hanya berdasarkan preferensi penulis saja. Penulis tidak memilih InternetExplor-  
505 eDriver, EdgeDriver, dan SafariDriver karena InternetExplorerDriver dan EdgeDriver hanya bisa  
506 berjalan pada sistem operasi Windows, selain itu SafariDriver hanya bisa berjalan pada sistem  
507 operasi macOS. HtmlUnitDriver tidak dipilih karena tidak mempunyai GUI sehingga tidak bisa  
508 dilakukan pengambilan *screenshot*. Pada subbab berikut dijelaskan beberapa kelas dan *interface*  
509 dari *library* Selenium WebDriver. Subbab ini mengacu pada [6].

### 510 2.3.1 Interface WebDriver

511 Merupakan *interface* utama yang digunakan untuk pengujian. Berikut ini adalah beberapa *method*  
512 dalam *interface* ini:

- 513 • void close()

514 Berfungsi untuk menutup *window* pada *browser*, jika *window* yang sekarang merupakan  
515 satu-satunya *window* yang terbuka maka *browser* akan ditutup.

- 516 • void quit()

517 Berfungsi untuk menutup *browser* dan semua *window* yang sedang terbuka.

- 518 • void get(String url)

519 Berfungsi untuk memuat halaman *web* pada *window* yang sedang aktif. *Method* ini mengirim  
520 *HTPP GET Request* untuk memuat halaman, dan *method* ini akan melakukan *blocking* sampai  
521 halaman *web* selesai dimuat.

522 Parameter: alamat *url* untuk memuat halaman *web*.

- 523 • String getTitle()

524 Berfungsi untuk mengembalikan judul dari halaman *web* yang sedang aktif.

525 Kembalian: judul dari halaman *web*.

- 526 • String getCurrentUrl()

527 Berfungsi untuk mendapatkan URL yang sedang aktif di *browser*.

528 Kembalian: URL dari halaman *web* yang sedang dimuat di *browser*.

- 529 • WebDriver.Options manage()

530 Berfungsi untuk mendapatkan interface WebDriver.Options.

531 Kembalian: interface dengan tipe WebDriver.Options.

532 **2.3.2 Interface WebDriver.Options**

533 *Interface* yang digunakan untuk mengatur *browser menu*. Berikut adalah *method* yang dimiliki oleh  
534 *interface* ini:

- 535 • void deleteAllCookies()

536 Berfungsi untuk menghapus semua *cookie* yang terdapat pada *browser*.

- 537 • WebDriver.Window window()

538 Berfungsi untuk mendapatkan *interface* untuk mengatur *browser window*.

539 Kembalian: objek bertipe WebDriver.Window.

540 **2.3.3 Interface WebDriver.Window**

541 *Interface* yang digunakan untuk mengatur *window* pada *browser*. Berikut adalah *method* yang  
542 dimiliki oleh *interface* ini:

- 543 • void maximize()

544 Berfungsi untuk membuat ukuran *window* pada *browser* menjadi maksimal.

545 **2.3.4 Kelas ChromeDriver**

546 Merupakan merupakan implementasi WebDriver yang mengontrol Chrome *browser*. Berikut adalah  
547 *constructor* yang terdapat pada kelas ini:

- 548 • public ChromeDriver()

549 *Constructor* ini membuat instans dengan tipe ChromeDriver.

550 **2.3.5 Kelas FirefoxDriver**

551 Merupakan merupakan implementasi WebDriver yang mengontrol Firefox *browser*. Berikut adalah  
552 *constructor* yang terdapat pada kelas ini:

- 553 • public FirefoxDriver()

554 *Constructor* ini membuat instans dengan tipe FirefoxDriver.

555 **2.3.6 Kelas OperaDriver**

556 Merupakan merupakan implementasi WebDriver yang mengontrol Opera *browser*. Berikut adalah  
557 *constructor* yang terdapat pada kelas ini:

- 558 • public OperaDriver()

559 *Constructor* ini membuat instans dengan tipe OperaDriver.

560 **2.3.7 Kelas SafariDriver**

561 Merupakan merupakan implementasi WebDriver yang mengontrol Safari *browser*. Berikut adalah  
562 *constructor* yang terdapat pada kelas ini:

- 563 • public SafariDriver()  
564     *Constructor* ini membuat instans dengan tipe SafariDriver.

565 **2.3.8 Kelas InternetExplorerDriver**

566 Merupakan merupakan implementasi WebDriver yang mengontrol Internet Explorer *browser*. Berikut  
567 adalah *constructor* yang terdapat pada kelas ini:

- 568 • public InternetExplorerDriver()  
569     *Constructor* ini membuat instans dengan tipe InternetExplorerDriver.

570 **2.3.9 Kelas EdgeDriver**

571 Merupakan merupakan implementasi WebDriver yang mengontrol Microsoft Edge *browser*. Berikut  
572 adalah *constructor* yang terdapat pada kelas ini:

- 573 • public EdgeDriver()  
574     *Constructor* ini membuat instans dengan tipe EdgeDriver.

575 **2.3.10 Kelas HtmlUnitDriver**

576 Merupakan merupakan implementasi WebDriver yang mengontrol HtmlUnit *browser*. Berikut  
577 adalah *constructor* yang terdapat pada kelas ini:

- 578 • public HtmlUnitDriver()  
579     *Constructor* ini membuat instans dengan tipe HtmlUnitDriver.

580 **2.3.11 Interface OutputType**

581 Merupakan *interface* yang menentukan tipe *output* pada *screenshot*. Terdapat tiga konstanta untuk  
582 menentukan tipe *output* pada *screenshot*. Konstanta tersebut adalah sebagai berikut:

- 583 • static final OutputType<String> BASE64  
584     Berfungsi untuk mendapatkan *screenshot* dalam bentuk *base64 data*.
- 585 • static final OutputType<byte[]> BYTES  
586     Berfungsi untuk mendapatkan *screenshot* dalam bentuk *raw bytes*.
- 587 • static final OutputType<java.io.File> FILE  
588     Berfungsi untuk mendapatkan *screenshot* dalam bentuk *temporary file* yang akan dihapus  
589 setelah program keluar dari *Java Virtual Machine*.

### 590 2.3.12 Interface TakesScreenshot

591 Merupakan *interface* yang digunakan untuk mengambil *screenshot*. WebDriver yang mengimple-  
592 mentasikan *interface* ini yaitu ChromeDriver, FirefoxDriver, OperaDriver, InternetExplorerDriver,  
593 dan EdgeDriver. HtmlUnitDriver tidak mengimplementasikan *interface* ini. Berikut adalah *method*  
594 yang terdapat dalam *interface* ini:

- 595 • <X> X getScreenshotAs(OutputType<X> target) throws WebDriverException  
596     *Method* ini berfungsi untuk mengambil *screenshot* dan mengembalikan hasilnya.  
597     Kembalian: objek hasil *screenshot*  
598     Parameter: tipe *output* yang diinginkan(lihat subbab [2.3.11](#)).

## 599 2.4 Apache Commons CLI

600 *Library* Apache Commons CLI menyediakan API untuk melakukan *parsing* argumen Command  
601 Line yang dikirimkan ke program[7]. Apache Commons CLI termasuk ke dalam salah satu *project*  
602 Apache Commons. Tujuan utama dari *project* Apache Commons adalah membuat dan melakukan  
603 *Maintain* pada komponen Java yang *reusable*. Pada subbab berikut dijelaskan beberapa kelas dan  
604 *interface* dari *library* Apache Commons CLI. Subbab ini mengacu pada [8].

### 605 2.4.1 Interface CommandLineParser

606 Merupakan sebuah *interface*. Kelas yang mengimplementasikan *interface* ini dapat melakukan  
607 *parsing* argumen CommandLine berdasarkan pada *option* yang telah ditentukan. Kelas-kelas yang  
608 mengimplementasikan *interface* ini yaitu BasicParser, GnuParser, PosixParser, dan DefaultParser.  
609 Kelas BasicParser, GnuParser, dan PosixParser sudah *deprecated*. Berikut ini adalah beberapa  
610 *method* yang dimiliki *interface* ini:

- 611 • CommandLine parse(Options options, String[] arguments) throws ParseException  
612     Berfungsi untuk melakukan *parsing* argumen Command Line berdasarkan pada *option* yang  
613     telah ditentukan. *Method* ini melempar *ParseException* jika terjadi masalah saat melakukan  
614     *parsing*.  
615     Parameter: *option* yang telah ditentukan, argumen Command Line.  
616     Kembalian: objek *CommandLine*.  
617 • CommandLine parse(Options options, String[] arguments, boolean stopAtNonOption) throws  
618     ParseException  
619     Berfungsi untuk melakukan *parsing* argumen Command Line berdasarkan pada *option* yang  
620     telah ditentukan.  
621     Parameter: *option* yang telah ditentukan, argumen Command Line, dan suatu *boolean* yang  
622     menentukan apakah *parsing* dihentikan jika terdapat *option* yang tidak valid. Jika bernilai  
623     *true*, *parsing* akan dihentikan dan semua argumen yang sudah diuraikan akan ditambahkan ke  
624     objek *CommandLine*. Jika bernilai *false*, akan dilempar *ParseException* bila terdapat *option*  
625     yang tidak valid.  
626     Kembalian: objek *CommandLine*.

**627 2.4.2 Kelas DefaultParser**

628 Merupakan kelas yang mengimplementasikan *interface* CommandLineParser. Berikut adalah *method*  
629 yang dimiliki kelas ini:

- 630 • CommandLine parse(Options options, String[] arguments) throws ParseException  
631 Berfungsi untuk melakukan *parsing* argumen Command Line berdasarkan pada *option* yang  
632 telah ditentukan. *Method* ini melempar *ParseException* jika terjadi masalah saat melakukan  
633 *parsing*.  
634 Parameter: *option* yang telah ditentukan, argumen Command Line.  
635 Kembalian: objek *CommandLine*.
- 636 • CommandLine parse(Options options, String[] arguments, boolean stopAtNonOption) throws  
637 ParseException  
638 Berfungsi untuk melakukan *parsing* argumen Command Line berdasarkan pada *option* yang  
639 telah ditentukan.  
640 Parameter: *option* yang telah ditentukan, argumen Command Line, dan suatu *boolean* yang  
641 menentukan apakah *parsing* dihentikan jika terdapat *option* yang tidak valid. Jika bernilai  
642 *true*, *parsing* akan dihentikan dan semua argumen yang sudah diuraikan akan ditambahkan ke  
643 objek *CommandLine*. Jika bernilai *false*, akan dilempar *ParseException* bila terdapat *option*  
644 yang tidak valid.  
645 Kembalian: objek *CommandLine*.

**646 2.4.3 Kelas CommandLine**

647 Kelas ini merepresentasikan kumpulan *option* hasil *parsing*. Berikut ini adalah beberapa *method*  
648 yang dimiliki kelas ini:

- 649 • public String getOptionValue(String opt)  
650 Berfungsi untuk mendapatkan nilai dari suatu *option* sesuai dengan namanya.  
651 Parameter: nama dari *option*.  
652 Kembalian: nilai dari suatu *option*. Jika *option* tidak ditemukan, akan mengembalikan *null*.
- 653 • public String[] getOptionValues(Option option)  
654 Berfungsi untuk mendapatkan kumpulan nilai dari suatu *option*. Parameter: nama dari *option*.  
655 Kembalian: kumpulan nilai dari suatu *option* dengan tipe *array of String*. Jika *option* tidak  
656 ditemukan, akan mengembalikan *null*.
- 657 • public Option[] getOptions()  
658 Berfungsi untuk mengembalikan *option* hasil *parsing*.  
659 Kembalian: *array* dari *option* hasil *parsing*.

**660 2.4.4 Kelas Options**

661 Kelas ini merepresentasikan kumpulan dari objek Option, yang mendeskripsikan kemungkinan  
662 *option* pada *command line*. Berikut ini adalah *method* yang dimiliki kelas ini:

- 663 • public Options addOption(Option opt)  
664 Berfungsi untuk menambahkan *option*.  
665 Parameter: Option yang akan ditambahkan.  
666 Kembalian: hasil dari *option* yang ditambahkan.

667 **2.4.5 Kelas Option**

668 Kelas ini merepresentasikan sebuah Command Line Option. Berikut ini adalah beberapa *method*  
669 yang dimiliki kelas ini:

- 670 • public String getLongOpt()  
671 Berfungsi untuk mendapatkan nama panjang dari suatu *option*.  
672 Kembalian: nama panjang dari suatu *option*.
- 673 • public String getValue()  
674 Berfungsi untuk mendapatkan nilai dari Option.  
675 Kembalian: nilai dari *option*. Jika terdapat lebih dari satu kembalian, hanya nilai pertama  
676 yang dikembalikan. Jika tidak ada nilai, akan mengembalikan *null*.
- 677 • public String[] getValues()  
678 Berfungsi untuk mendapatkan nilai-nilai dari *option*.  
679 Kembalian: nilai-nilai dari *option* berupa *array of String*. Jika tidak ada nilai, akan mengem-  
680 balikan *null*.
- 681 • public boolean hasArg()  
682 Berfungsi untuk mengetahui apakah suatu *option* membutuhkan argumen.  
683 Kembalian: *true* jika *option* ini membutuhkan argumen , *false* jika *option* ini tidak membu-  
684 tuhkan argumen.
- 685 • public boolean hasArg()  
686 Berfungsi untuk mengetahui apakah suatu *option* dapat menerima lebih dari satu argumen.  
687 Kembalian: *true* jika *option* ini dapat menerima lebih dari satu argumen , *false* jika tidak.
- 688 • public String getDescription()  
689 Berfungsi untuk mendapatkan deskripsi dari suatu *option*.  
690 Kembalian: deskripsi dari *option* ini.
- 691 • public String getArgName()  
692 Berfungsi untuk mendapatkan nama dari suatu *option*.  
693 Kembalian: nama dari argumen suatu *option*

694 **2.4.6 Kelas Option.Builder**

695 Kelas ini merupakan *nested class* dari kelas Option. Kelas ini digunakan untuk membuat objek  
696 Option berdasarkan parameter yang diberikan. Berikut ini adalah beberapa *method* yang dimiliki  
697 kelas ini:

- 698 • public Option.Builder desc(String description)  
699 Berfungsi untuk memberikan deskripsi pada *option*.  
700 Parameter: deskripsi dari *option*.  
701 Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- 702 • public Option.Builder longOpt(String longOpt)  
703 Berfungsi untuk memberikan nama panjang pada *option*.  
704 Parameter: nama panjang *option*.  
705 Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- 706 • public Option.Builder hasArg()  
707 Berfungsi untuk menyatakan bahwa *option* ini membutuhkan argumen.  
708 Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- 709 • public Option.Builder hasArgs()  
710 Berfungsi untuk menyatakan bahwa *option* ini membutuhkan argumen, dimana jumlah  
711 argumen bisa lebih dari satu.  
712 Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- 713 • public Option.Builder argName(String argName)  
714 Berfungsi untuk memberi nama pada argumen.  
715 Parameter: nama argumen.  
716 Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- 717 • public Option.Builder required(boolean required)  
718 Berfungsi untuk menyatakan bahwa *option* ini wajib ada.  
719 Parameter: variabel bertipe *boolean* yang menentukan apakah *option* ini wajib ada.  
720 Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- 721 • public Option build()  
722 Berfungsi untuk membuat objek Option berdasarkan nilai pada kelas Option.Builder.  
723 Kembalian: objek Option.



724

## BAB 3

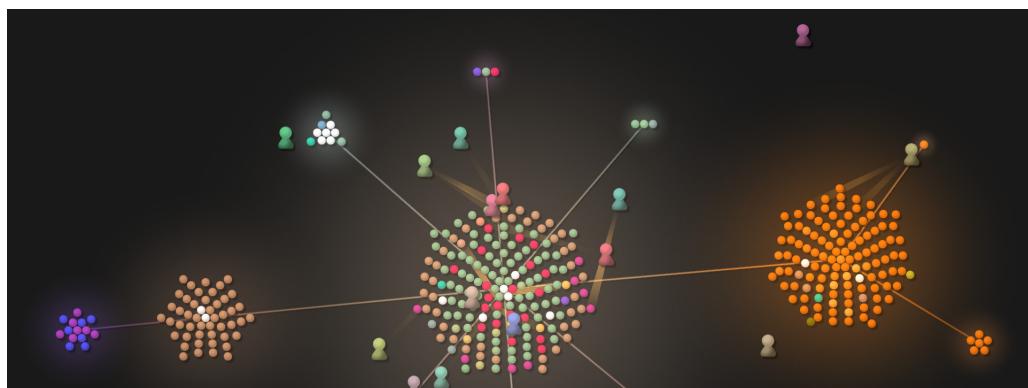
725

## ANALISIS

726 Pada bab ini dijelaskan mengenai analisis aplikasi sejenis, analisis penggunaan JGit dan Selenium  
727 WebDriver untuk membangkitkan animasi timelapse, prapengujian, dan analisis fitur aplikasi yang  
728 dibangun.

### 729 3.1 Analisis Aplikasi Sejenis

730 Saat skripsi ini dibuat, aplikasi sejenis yang digunakan untuk membangkitkan animasi adalah Gource.  
731 Proyek perangkat lunak ditampilkan oleh Gource sebagai animasi pohon, dimana pusatnya adalah  
732 *root directory* dari proyek perangkat lunak[9]. Direktori ditampilkan sebagai *branch*, sedangkan *file*  
733 ditampilkan sebagai *leaf*. Developer dapat terlihat di *working tree* pada saat mereka berkontribusi  
734 untuk proyek.



Gambar 3.1: Visualisasi proyek perangkat lunak menggunakan Gource.

735 Gambar 3.1 menunjukkan contoh visualisasi proyek perangkat lunak menggunakan Gource. Efek  
736 cahaya yang terdapat pada Gambar 3.1 disebut dengan *bloom*. Pada awalnya ukuran *working tree*  
737 tidak terlalu besar. Setiap kali ditambahkan *file* dan *folder* baru, akan dibuat *branch* dan *leaf* baru  
738 pada *working tree*.

739

740 Gource memiliki beberapa fitur. Fitur-fitur tersebut dapat diatur melalui Command Line Option.  
741 Berikut ini adalah beberapa Command Line Option yang terdapat pada Gource:

742 1. `gource -<WIDTH>x<HEIGHT>`

743 Opsi ini berfungsi untuk mengatur resolusi layar dari animasi. Parameter dari opsi ini adalah  
744 lebar dan panjang layar dalam satuan piksel.

**745 2. gource -camera-mode <MODE>**

746 Opsi ini berfungsi untuk mengatur mode kamera pada Gource. Parameter dari opsi ini adalah  
747 mode dari kamera. Terdapat dua mode yaitu *overview* dan *track*. Dalam mode *track*, kamera  
748 bergerak mengikuti *user* yang sedang aktif. Dalam mode *overview*, kamera menampilkan  
749 seluruh repositori.

**750 3. gource -path <PATH>**

751 Opsi ini berfungsi untuk mengatur *path* dari direktori yang akan dibuat animasinya. Opsi  
752 dari parameter ini adalah *path* dari direktori.

**753 4. gource -start-date <YYYY-MM-DD hh:mm:ss +tz> -stop-date <YYYY-MM-DD hh:mm:ss +tz>**

754 Opsi ini berfungsi untuk mengatur periode waktu dalam menampilkan animasi. Parameter  
755 dari opsi ini adalah waktu mulai dan waktu akhir dalam format "YYYY-MM-DD hh:mm:ss  
756 +tz". Dimana YYYY adalah tahun, MM adalah bulan, DD adalah tanggal, hh adalah jam,  
757 mm adalah menit, ss adalah detik, dan +tz adalah zona waktu. Parameter jam, menit, detik,  
758 dan zona waktu bersifat opsional.

**759 5. gource -bloom-multiplier <FLOAT>**

760 Opsi ini berfungsi untuk mengatur radius dari efek *bloom*. Parameter dari opsi ini adalah  
761 radius dalam format bilangan riil.

**762 6. gource -bloom-intensity <FLOAT>**

763 Opsi ini berfungsi untuk mengatur intensitas dari efek *bloom*. Parameter dari opsi ini adalah  
764 intensitas *bloom* dalam format bilangan riil.

**765 7. gource -disable-bloom**

766 Opsi ini berfungsi untuk menonaktifkan animasi *bloom*.

**767 8. gource -date-format <FORMAT>**

768 Opsi untuk mengatur format waktu yang ditampilkan pada bagian tengah atas. Opsi dari  
769 parameter ini adalah format waktu dalam bentuk *string*.

**770 9. gource -background <FFFFFF>**

771 Opsi ini berfungsi untuk mengatur warna *background*. Parameter dari opsi ini adalah warna  
772 dalam format heksadesimal.

**773 10. gource -background-image <IMAGE>**

774 Opsi ini berfungsi untuk mengatur gambar *background*. Parameter dari opsi ini adalah nama  
775 file dari gambar.

**776 11. gource -font-size <SIZE>**

777 Opsi ini digunakan untuk mengatur ukuran *font* pada tulisan *title* dan tanggal. Parameter  
778 dari opsi ini adalah ukuran *font*.

**779 12. gource -font-colour <FFFFFF>**

780 Opsi ini digunakan untuk mengatur warna *font* pada tulisan *title* dan tanggal. Parameter  
781 dari opsi ini adalah warna *font* dalam format heksadesimal.

782 13. `gource -logo <IMAGE>`

783 Opsi ini berfungsi untuk memasukkan logo. Parameter dari opsi ini adalah nama *file* dari  
784 gambar.

785 14. `gource -logo-offset <X>x<Y>`

786 Opsi ini berfungsi untuk mengatur posisi dari logo. Parameter dari opsi ini adalah posisi x  
787 dan posisi y dari logo.

788 15. `gource -title <TITLE>`

789 Opsi ini berfungsi untuk memberi judul. Dimana judul tersebut ditampilkan pada pojok kiri  
790 bawah layar.

791 16. `gource -output-framerate <FPS>`

792 Opsi ini berfungsi untuk mengatur jumlah *frame* per detik pada video animasi. Parameter  
793 dari opsi ini adalah jumlah *frame* per detik.

794 17. `gource -seconds-per-day <SECONDS>`

795 Opsi ini berfungsi untuk mengatur kecepatan visualisasi. Parameter dari opsi ini adalah waktu  
796 yang dibutuhkan untuk menampilkan satu hari.

797 18. `gource -hide <DISPLAY-ELEMENT>`

798 Opsi ini berfungsi untuk menyembunyikan satu atau lebih *display element*. Parameter dari  
799 opsi ini adalah elemen yang akan disembunyikan. *Display element* yang dapat disembunyikan  
800 yaitu:

- 801 • *bloom*: efek *bloom*.
- 802 • *date*: waktu.
- 803 • *dirnames*: nama direktori.
- 804 • *files*: ikon dari berkas.
- 805 • *filenames*: nama berkas.
- 806 • *root*: *root directory*.
- 807 • *users*: ikon dari *user*.
- 808 • *usernames*: nama dari *user*.

809 Parameter yang berjumlah lebih dari satu dipisahkan dengan koma, contoh: *bloom,root,users*.

810 Gource dapat digunakan untuk berbagai macam proyek perangkat lunak. Program pada skripsi

811 ini hanya akan berfokus untuk proyek perangkat lunak berbasis *web*. Tidak seperti Gource yang  
812 menampilkan direktori dan *file* pada animasi, program pada skripsi ini menampilkan *screenshot*  
813 dari halaman suatu *website*.

## 814 3.2 Analisis Penggunaan JGit dan Selenium WebDriver

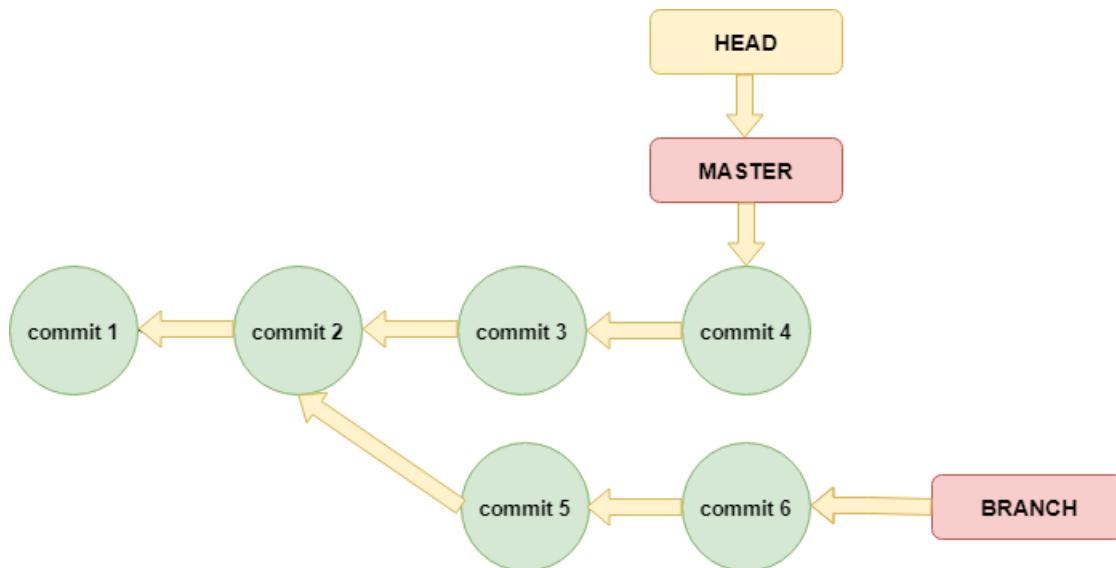
815 Pada subbab ini dijelaskan mengenai analisis penggunaan JGit dan Selenium WebDriver.

### 3.2.1 Analisis Penggunaan JGit

JGit dapat digunakan untuk berinteraksi dengan repositori yang terekam oleh Git. Dimana interaksi ini dapat dilakukan melalui program java. Pada analisis ini dibahas kelas-kelas pada *library* JGit yang digunakan pada program dalam skripsi ini. Kelas-kelas yang dipakai yaitu Git, Repository, FileRepository, dan RevCommit.

Untuk dapat berinteraksi dengan suatu repositori Git diperlukan kelas Repository dan Git. Kelas Repository dan FileRepository merepresentasikan suatu repositori Git. Kelas FileRepository merupakan turunan dari kelas Repository. Operasi-operasi pada Git dapat dilakukan dengan menggunakan kelas Git. *Constructor* dari kelas Git menerima parameter bertipe Repository. Repository bersifat abstrak, karena itu tidak bisa diinisialisasi secara langsung. Repository dapat diinisialisasi menggunakan *object* yang bertipe FileRepository. Dimana *constructor* FileRepository menerima parameter berupa alamat dari direktori Git.

Dibutuhkan beberapa langkah untuk melakukan operasi Git Log. Dari object bertipe Git, dapatkan object bertipe LogCommand dengan memanggil *method* log(). Setelah itu, panggil *method* call() untuk melakukan operasi Git Log. Setelah operasi Git Log dijalankan, akan didapatkan seluruh *commit* pada *branch* yang sedang aktif, dimana seluruh *commit* tersebut berupa *object* yang bertipe Iterable<RevCommit>. Jika histori seperti pada Gambar 3.2 dan saat ini HEAD berada di Master, Iterable<RevCommit> akan berisi commit4, commit3, commit2, dan commit1. Jika saat ini HEAD berada di BRANCH maka Iterable<RevCommit> akan berisi commit6, commit5, commit2, dan commit1. Operasi Git Log ini nantinya dipakai oleh program dalam skripsi ini untuk mendapatkan seluruh histori *commit* dari perangkat lunak berbasis *web*.



Gambar 3.2: Histori *commit* direpresentasikan sebagai Directed Acyclic Graph.

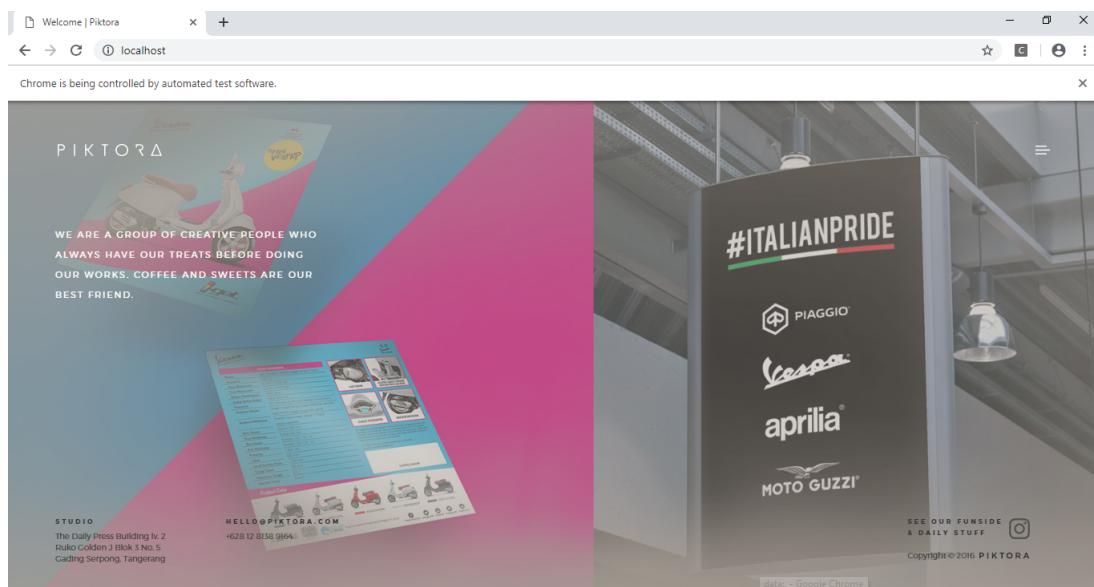
Dibutuhkan beberapa langkah untuk melakukan operasi Git Checkout. Dari object bertipe Git, dapatkan object bertipe CheckoutCommand dengan memanggil *method* checkout(). Setelah itu, panggil *method* setName() diikuti dengan parameter berupa *commit ID*. Kemudian panggil *method* call() untuk menjalankan operasi Git Checkout. Setelah operasi Git Checkout dijalankan, *working directory* akan diperbarui sesuai dengan *state* pada *commit* tertentu. Operasi *checkout* ini nantinya

842 akan digunakan oleh program dalam skripsi ini untuk menelusuri *commit* dan mendapatkan halaman  
 843 *web* dari perangkat lunak berbasis *web*.

### 844 3.2.2 Analisis Penggunaan Selenium WebDriver

845 Selenium WebDriver dapat digunakan untuk mengotomatisasi *web browser*. Pada analisis ini dibahas  
 846 kelas dan *interface* pada *library* Selenium WebDriver yang digunakan dalam program pada skripsi  
 847 ini. Kelas yang dipakai yaitu ChromeDriver, sedangkan *interface* yang dipakai adalah WebDriver  
 848 dan TakeScreenshot.

849 WebDriver merupakan *interface* utama yang digunakan untuk pengujian. WebDriver dapat  
 850 diinisialisasi menggunakan *object* dengan tipe ChromeDriver. Setelah melakukan inisialisasi pada  
 851 WebDriver, *browser* akan dijalankan. Untuk membuka suatu halaman *web*, digunakan *method*  
 852 `get()` dengan parameter alamat URL. Gambar 3.3 menunjukkan Chrome *browser* yang dikontrol  
 853 oleh ChromeDriver. WebDriver ini nantinya akan digunakan oleh program dalam skripsi ini untuk  
 854 membuka halaman *web* pada perangkat lunak berbasis *web*.



Gambar 3.3: *Browser* yang dikontrol oleh ChromeDriver.

855 TakesScreenshot merupakan *interface* yang digunakan untuk menangkap *screenshot* halaman  
 856 *web* pada WebDriver. Kelas ChromeDriver mengimplementasikan *interface* ini. *Method* yang  
 857 digunakan untuk menangkap *screenshot* adalah `getScreenshotAs()`, diikuti dengan parameter bertipe  
 858 *OutputType*. *OutputType* ini merupakan *interface* yang digunakan untuk menentukan tipe *output*  
 859 dari hasil *screenshot*. *OutputType* yang digunakan pada program dalam skripsi ini adalah *File*.  
 860 Tipe ini dipilih karena memudahkan kelas *ImageIO*<sup>1</sup> dalam membaca *image*. Pada *OutputType*  
 861 *BYTES*, byte harus dibaca dulu oleh kelas *ByteArrayInputStream*<sup>2</sup>, kemudian dibaca oleh kelas  
 862 *ImageIO*. Pada *OutputType* *BASE64*, *base64 data* dikonversi menjadi *array of bytes*, setelah itu  
 863 dibaca oleh *ByteArrayInputStream*, kemudian dibaca oleh kelas *ImageIO*. Pada *OutputType* *FILE*,  
 864 *file* dapat langsung dibaca oleh kelas *ImageIO*. TakesScreenshot ini nantinya akan digunakan oleh

<sup>1</sup><https://docs.oracle.com/javase/7/docs/api/javax/imageio/ImageIO.html>

<sup>2</sup><https://docs.oracle.com/javase/7/docs/api/java/io/ByteArrayInputStream.html>

865 program dalam skripsi ini untuk mengambil *screenshot* dari halaman *web* pada perangkat lunak  
866 berbasis *web*.

### 867 3.3 Analisis Fitur Aplikasi yang Dibangun

868 Pada skripsi ini, dibuat sebuah perangkat lunak yang dapat membangun animasi *timelapse* dari  
869 pengembangan proyek perangkat lunak berbasis web. Yang akan dibuat animasinya adalah halaman  
870 web dari perangkat lunak. Jumlah halaman berkisar antara satu sampai empat halaman, tergantung  
871 pada masukan dari *user*. *Output* dari program adalah *file* hasil animasi yang bertipe GIF. Program  
872 dapat menerima masukan dan konfigurasi dari Command Line Option. Setiap *option* harus memiliki  
873 argumen.

874 Berikut ini adalah *option* yang akan diimplementasikan pada skripsi ini:

875 1. **-project-path <PATH>**

876 Opsi ini berfungsi untuk mengatur *path* dari direktori yang akan dibuat animasinya. Parameter  
877 dari opsi ini adalah *path* dari proyek perangkat lunak web yang terekam oleh Git. Opsi ini  
878 wajib ada.

879 2. **-before-capture <TERMINAL-COMMAND>**

880 Opsi ini berfungsi untuk menjalankan *terminal command*. *Terminal command* ini dijalankan  
881 sebelum melakukan *screenshot*. Parameter dari opsi ini adalah *terminal command* dalam  
882 bentuk *string* dan harus diapit dengan tanda petik ganda. Opsi ini bersifat opsional. Jika  
883 opsi ini tidak dimasukkan, program tidak akan menjalankan *terminal command*.

884 3. **-capture-url <URL 1>[, URL 2[, URL 3[, URL 4]]]**

885 Opsi ini berfungsi untuk mengatur alamat *url* dari halaman *web*, dimana dilakukan pengambil-  
886 an *screenshot* pada halaman ini. Jumlah halaman yang dicapture bisa lebih dari satu, dengan  
887 jumlah maksimal empat halaman. Parameter dari opsi ini adalah alamat *url* dari halaman  
888 *web*, dengan jumlah maksimal empat alamat *url*. Opsi ini wajib ada.

889 4. **-seconds-per-commit <SECONDS>**

890 Opsi ini berfungsi untuk mengatur durasi munculnya satu *commit* pada animasi. Parameter  
891 dari opsi ini adalah durasi munculnya satu *commit* dalam satuan detik. Parameter harus  
892 berupa bilangan bulat atau riil, dimana nilainya lebih besar dari nol dan kurang dari 656.  
893 Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, nilai defaultnya adalah 1 detik.

894 5. **-title <TITLE>**

895 Opsi ini berfungsi untuk memberi judul. Dimana judul tersebut ditampilkan pada pojok kiri  
896 bawah layar. Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, maka judul tidak  
897 akan ditampilkan.

898 6. **-logo <IMAGE>**

899 Opsi ini berfungsi untuk memasukkan logo. Dimana logo tersebut ditampilkan pada pojok  
900 kanan bawah layar. Parameternya adalah *path* dari *file* gambar. Format gambar yang  
901 diterima antara lain JPG, PNG, GIF dan, BMP. Opsi ini bersifat opsional. Jika opsi ini tidak  
902 dimasukkan, maka judul tidak akan ditampilkan.

903 7. **-start-commit <COMMIT-ID>**

904 Opsi ini berfungsi untuk mengatur *commit* awal untuk membangkitkan animasi. Parameter  
905 dari opsi ini adalah *commit ID* awal , dimana panjang *commit ID* adalah tujuh sampai  
906 sepuluh karakter. Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, program akan  
907 membangkitkan animasi dimulai dari *commit* paling awal.

908 8. **-stop-commit <COMMIT-ID>**

909 Opsi ini berfungsi untuk mengatur *commit* akhir untuk membangkitkan animasi. Parameter  
910 dari opsi ini adalah *commit ID* akhir , dimana panjang *commit ID* adalah tujuh sampai  
911 sepuluh karakter. Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, program akan  
912 membangkitkan animasi sampai dengan *commit* paling akhir.

913 9. **-branch <BRANCH>**

914 Opsi ini berfungsi untuk mengatur *branch* dari repositori yang akan dibangkitkan animasinya.  
915 Parameter dari opsi ini adalah nama dari *branch*. Opsi ini bersifat opsional. Jika opsi ini  
916 tidak dimasukkan, program akan membangkitkan animasi pada *branch* master.

917

918 Opsi **-path**, **-title**, dan **-image** mengacu pada opsi yang terdapat pada Gource. Opsi **-start-commit**  
919 **-stop-commit** dan **-seconds-per-commit** mengacu pada Gource, dengan sedikit penyesuaian.

920 Opsi **-seconds-per-day** pada Gource menyatakan durasi munculnya satu hari, sedangkan Opsi

921 **-seconds-per-commit** menyatakan durasi munculnya satu *commit*. Opsi **-start-date** **-stop-date**

922 pada Gource dan opsi **-start-commit** **-stop-commit** mengatur periode dalam menampilkan ani-

923 masi. Pada Gource, periode yang digunakan adalah rentang waktu berupa tanggal, bulan, dan

924 tahun. Sedangkan pada opsi **-start-commit** **-stop-commit**, rentangnya berupa *commit ID*. Opsi

925 **-before-capture**, **-capture-url**, dan **-branch** tidak mengacu pada Gource. Ketiga opsi tersebut

926 secara khusus dibuat karena program dalam skripsi ini membuat animasi *timelapse* dari kumpulan

927 halaman *web*, dimana halaman *web* tersebut diambil dari proyek perangkat lunak berbasis *web* yang

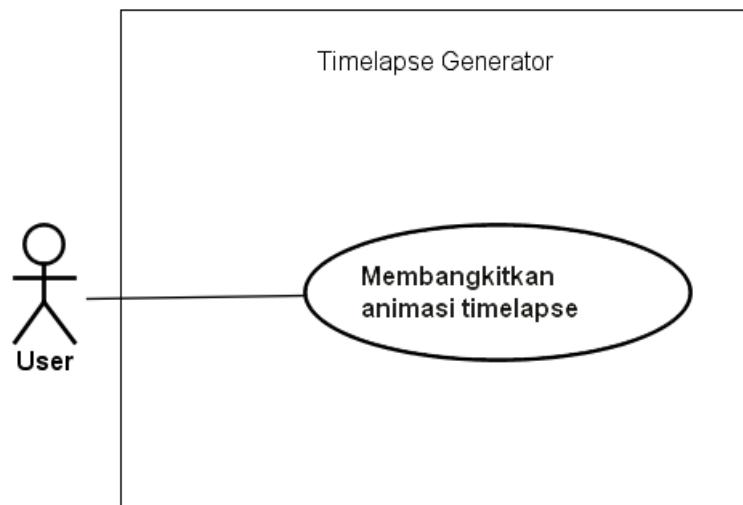
928 terekam oleh Git. Sedangkan program Gource membuat animasi dari struktur *folder* atau *file*.

929 Program pada skripsi ini memiliki satu fitur. Fitur tersebut adalah membangkitkan animasi

930 *timelapse*. Penjelasan fitur dapat dilihat pada Gambar 3.4 dan Tabel 3.1.

|             |   |
|-------------|---|
| Nama        | Membangkitkan animasi <i>timelapse</i>  |
| Deskripsi   | Fitur untuk membangkitkan animasi timelapse dari pengembangan proyek perangkat lunak berbasis web   |
| Aktor       | <i>User</i>   |
| Pos-kondisi | File animasi bertipe GIF berhasil dibuat  |
| Skenario    | <ol style="list-style-type: none"> <li>1. Sistem membaca masukan <i>input</i> dari Command Line Option.</li> <li>2. Sistem menelusuri histori perkembangan perangkat lunak berbasis <i>web</i> dengan fitur Git. Saat menelusuri histori perkembangan perangkat lunak, sistem mengambil <i>screenshot</i> dari halaman web menggunakan SeleniumWebDriver.</li> <li>3. Sistem menggabungkan file <i>screenshot</i> menjadi satu file bertipe GIF.</li> </ol> |

Tabel 3.1: *Scenario case* membangkitkan animasi *timelapse*



Gambar 3.4: *Use case diagram* perangkat lunak.

931

932

933 Langkah-langkah untuk membangkitkan animasi *timelapse* adalah sebagai berikut:

934

- 935 1. Program membaca argumen Command Line menggunakan Apache Commons CLI.
- 936 2. Program mendapatkan seluruh histori *commit* dari repositori perangkat lunak berbasis web pada menggunakan JGit. Jika terdapat *option -branch*, program akan mendapatkan histori

938        *commit* pada suatu *branch*. Jika tidak terdapat *option -branch*, program akan mendapatkan  
939        seluruh histori *commit* pada *branch* master.

- 940        3. Program membuka semua *browser* menggunakan Selenium WebDriver. Jumlah *browser*  
941        bergantung pada jumlah argumen dari *option -capture-url*.
- 942        4. Program melakukan *checkout* dalam suatu rentang *commit* tertentu. Jika tidak terdapat  
943        *option -start-commit* dan *option -stop-commit*, akan dilakukan *checkout* ke semua *commit*.
- 944        5. Program melakukan *checkout* ke *commit* awal dari rentang *commit*.
- 945        6. Program menjalankan *terminal Command* jika terdapat *option -before-capture*.
- 946        7. Program membuka setiap URL yang didapatkan dari *option -capture-url* menggunakan  
947        Selenium WebDriver. Setiap *browser* membuka URL yang berbeda.
- 948        8. Selenium WebDriver kemudian mengambil *screenshot* pada semua *browser*.
- 949        9. Ulangi langkah 5-8 sampai semua *commit* yang berada di rentang *commit* diambil *screenshot*  
950        halamannya.
- 951        10. Jika terdapat lebih dari satu *browser*, hasil *screenshot* dari setiap *browser* digabungkan menjadi  
952        satu gambar baru.
- 953        11. Program menambahkan judul di pojok kiri bawah jika terdapat *option -title*.
- 954        12. Program menambahkan logo di pojok kanan bawah jika terdapat *option -logo*.
- 955        13. Program menggabungkan semua *file* gambar menjadi satu *file* bertipe GIF.

956        Penggabungan *screenshot* halaman-halaman *web* menjadi *file* GIF dilakukan dengan bantuan  
957        *library*<sup>3</sup>.

## 958 3.4 Prapengujian Website Piktora

959        Pengujian dilakukan dengan proyek Piktora sebagai input dari program. Piktora<sup>4</sup> merupakan situs  
960        *web* yang menawarkan layanan *creative design* dan *branding*. Layanan yang ditawarkan berupa  
961        *graphic design* untuk poster, *banner*, *website*, dan aplikasi *mobile*. Reposisori dari situs *web* ini  
962        disimpan pada Gitlab<sup>5</sup>, dibutuhkan akses khusus untuk membuka repositori tersebut. Piktora  
963        dibangun dengan php menggunakan framework Code Igniter. Piktora menggunakan MySQL sebagai  
964        basis datanya.

965        Prapengujian ini hanya menggunakan dua parameter, yaitu project-path dan capture-url(lihat  
966        subbab 3.3). Pengujian dilakukan menggunakan prototipe program yg akan dilengkapi pada Bab 5.  
967        Berikut adalah spesifikasi perangkat keras dan perangkat lunak yang digunakan untuk prapengujian:

- 968        • Processor: Intel Core i3 4030U

<sup>3</sup><http://elliot.kroo.net/software/java/GifSequenceWriter>

<sup>4</sup><http://www.piktora.com>

<sup>5</sup><https://gitlab.com/PNDevworks/Piktora>

- 969 • RAM: 6GB
- 970 • Sistem Operasi: Windows 10 pro 64-bit
- 971 • Versi Apache HTTP Server: 2.4.29
- 972 • Versi MySQL Server: 5.5.5
- 973 • Versi Netbeans: 8.1
- 974 • Versi Google Chrome: 73.0.3683.86

975 Apache HTTP Server digunakan sebagai *web server* lokal. MySQL Server digunakan untuk  
 976 menyimpan basis data secara lokal dan juga digunakan sebagai *database server* lokal.

977 Pada proyek Piktora terdapat 58 *commit*. Listing 3.1 menunjukkan histori *commit* pada proyek  
 978 Piktora, histori didapatkan dengan menjalankan operasi Git Log pada terminal. Setelah dilakukan  
 979 pengujian terdapat beberapa masalah. Masalah tersebut yaitu perbedaan letak *file*, migrasi *database*,  
 980 dan konfigurasi *database*. Masalah-masalah tersebut dibahas pada subbab 3.4.1 sampai 3.4.3.

Listing 3.1: Histori *commit* pada proyek Piktora

```
981 315d374 - Oct 31 16:52:46 2016 : Basic CI files + htaccess & webconfig + database .
982     php ignore
983
984 27ce3d4 - Nov 5 13:12:43 2016 : setup environment for piktora
985 65f0c9c - Nov 5 19:22:58 2016 : * create structure for all pages * add dummy
986     images
987
988 bffbae1 - Nov 8 18:00:32 2016 : - basic structure (navbar semi complete) - add
989     fonts
990
991 5c59916 - Nov 8 19:51:18 2016 : implement navbar, footer, and projects/ page
992 7738380 - Nov 8 20:05:27 2016 : fix pc and ipad navbar fontsize
993 26bdbbe - Nov 8 20:16:33 2016 : fix position image for desktop /projects
994 3db3ce8 - Nov 9 00:28:27 2016 : - implements project details page - fix some minor
995     issue - add some project image
996
997 5ef34fa - Nov 13 13:01:06 2016 : implement about us (raw version)
998 3caf535 - Nov 15 11:55:15 2016 : fix minor issues view/about: - background-image
999     position. Make it to the center position - slick.js img need to set to inline
1000
1001 c5eb3b6 - Nov 15 13:02:42 2016 : implement /welcome page
1002 ade9216 - Nov 15 13:12:08 2016 : fix minor issues: - move style footer to global -
1003     add space before PIKTORA
1004
1005 c77b5b3 - Nov 18 18:18:25 2016 : implement /contact
1006 b42b819 - Nov 18 21:22:22 2016 : change a href to style cursor: pointer
1007 3eb7af8 - Nov 21 16:09:40 2016 : .htaccess to be compatible with cloud kilat
1008 e87e84b - Nov 22 14:53:45 2016 : - change vw to 100% - add captcha
1009 ff8d829 - Nov 22 15:22:53 2016 : Solved captcha font load: use otf instead of ttf
1010     Also: create directory assets/img/captcha and ignore everything inside
1011
1012 dc87342 - Nov 22 15:49:03 2016 : - implement captcha code - remove wrong css
```

1026 | 9ebe433 – Nov 22 16:12:44 2016 : add scroll feature in project/  
1027 | f0f7270 – Nov 23 15:17:42 2016 : Added Google PHP Client v2 See <https://github.com/google/google-api-php-client>  
1028 |  
1029 | 57a239b – Nov 23 15:19:57 2016 : Merge origin/master  
1030 | e2dfebe – Nov 27 14:42:24 2016 : – remove blue outline when click with slick –  
1019 | change background image in about to newest one  
1031 |  
1032 | a19e7f2 – Nov 27 15:24:42 2016 : Detailing dari Edina: 1. Hal. Project Detail ,  
1022 | font coba diperkecil saja mungkin ya. 3. Beberapa ukuran font dan spacing ada  
1023 | yang kurang pas sedikit , terlampir detail revisinya ya (file pdf) 4. Footer  
1024 | dibuat selalu stay terus di bagian bawah dengan posisi yang selalu sama. Di home  
1025 | & contact sudah sama , namun di hal. product posisinya agak lebih naik.  
1033 |  
1034 | 3d79d0a – Nov 27 15:29:58 2016 : fix minor issue  
1035 | 0fc958 – Nov 28 10:11:06 2016 : add raw admin contents  
1036 | add3974 – Nov 28 12:03:21 2016 : add summernote , implement read project  
1037 | 0680488 – Nov 29 12:38:23 2016 : update admin for projects  
1038 | fbe7639 – Nov 29 13:10:30 2016 : implement admin for home  
1039 | db0cedd – Nov 29 13:38:29 2016 : – implementasi database bagian user – upload 9  
1033 | gambar contoh project  
1040 |  
1041 | 0fe9aaaf – Nov 29 14:17:20 2016 : ubah warna garis captcha  
1042 | f2326dd – Nov 29 14:44:51 2016 : – lewati proses otentikasi sementara  
1043 | f78cdb4 – Dec 2 12:10:47 2016 : (trying to) fix issue #2  
1044 | ef9b62b – Dec 2 17:09:58 2016 : revisi dari edina ke-2  
1045 | c689aa8 – Dec 2 17:11:13 2016 : perbaikan admin sedikit  
1046 | c4e9576 – Dec 2 17:14:06 2016 : perbaikan di /contact , kelewat  
1047 | 02d04f1 – Dec 5 14:55:20 2016 : tambah wording  
1048 | a4e4858 – Dec 5 15:08:59 2016 : perbaikan kata2 sedikit  
1049 | bbd82c2 – Dec 6 10:41:40 2016 : implementasi email  
1050 | f8c64fc – Dec 6 11:03:08 2016 : change to httpdocs  
1051 | eb49c2b – Dec 6 11:35:20 2016 : hapus migrasi script di admin  
1052 | ace1988 – Dec 6 11:39:00 2016 : change overflow to auto  
1053 | 627e65b – Dec 6 11:45:26 2016 : modify database back to local  
1054 | 0896f81 – Dec 7 16:08:30 2016 : update home versi mobile jadi baru (revisi dari  
1049 | Edina)  
1055 |  
1056 | 5cf1292 – Dec 7 16:21:01 2016 : ubah background di about menjadi tidak pecah  
1057 | c83f4aa – Thu Dec 15 15:04:30 2016 : remove piktora secrets  
1058 | 57f5ea4 – Thu Dec 15 15:09:43 2016 : remove unimportant data  
1059 | 7931c21 – Dec 24 18:40:41 2016 : edit wording  
1060 | 9b0a302 – Dec 25 06:03:50 2016 : Another wording fix  
1061 | flea410 – Thu Jan 5 15:23:32 2017 : fix instagram link  
1058 | 1880a88 – Thu Jan 5 15:24:12 2017 : Merge branch 'master' of <https://github.com/pascalalfadian/Piktora>  
1063 |  
1064 | 286aa78 – Jan 16 12:48:45 2017 : Perbaikan wording di admin edit project  
1065 | 33702c2 – Feb 21 13:31:08 2017 : change email sender to piktora@mailgun.dnartworks.  
1062 | com.au  
1066 |  
1067 | 18c39ef – Thu Apr 13 15:21:49 2017 : Test commit (in gitlab). Nothing much important

```

168 | 9bfde3c - Apr 17 15:09:54 2017 : add ignore sftp-config.json
169 | 38711f0 - Apr 17 15:15:03 2017 : fix bug ugly display when projects too high
170 | 9f041ef - May 15 10:40:16 2017 : set insta url to https://www.instagram.com/
1068 |     piktorastudio/
1069 |
1070 | 6a085c1 - Dec 12 14:38:38 2017 : Update company address
1071 | 89000be - Jan 12 12:25:30 2018 : Update new company address

```

### 1072    3.4.1 Perbedaan Letak *File*

1073 Pada *commit* 315d374 (31 Oktober 2016) s.d. bbd82c2 (6 Desember 2016) halaman *web* proyek  
 1074 Piktora tidak bisa dibuka. Hal ini disebabkan oleh perbedaan letak *file* "index.php". Pada *commit*  
 1075 315d374 (31 Oktober 2016) s.d. bbd82c2 (6 Desember 2016), *file* "index.php" berada pada *folder*  
 1076 "www", sedangkan pada *commit* f8c64fc (6 Desember 2016) s.d. 89000be (12 Januari 2018) *file*  
 1077 "index.php" berada pada *folder* "httpdocs". Akibat adanya perbedaan letak *file* tersebut, maka  
 1078 konfigurasi dari apache harus diubah.

1079 Solusi untuk mengatasi masalah ini adalah dengan menambahkan *option* yang akan menjalankan  
 1080 *terminal command*. Dimana *terminal command* ini berisi perintah untuk menjalankan *script*  
 1081 PHP. *Script* PHP ini akan mengecek letak *file* "index.php" pada *folder* "www" dan "httpdocs".  
 1082 *Script* kemudian akan mengecek *directory root* apache pada *file* "httpd.conf". Jika *directory root*  
 1083 sudah mengarah ke *folder* tempat "index.php" berada, maka *script* tidak akan mengubah isi *file*  
 1084 "httpd.conf". Jika *directory root* tidak mengarah ke *folder* tempat "index.php" berada, maka *script*  
 1085 akan mengubah *directory root* pada *file* "httpd.conf" dan melakukan *restart* pada apache. Setelah  
 1086 itu *script* akan melakukan migrasi *database*.

### 1087    3.4.2 Permasalahan Konfigurasi *Database*

1088 Pada *commit* f8c64fc (6 Desember 2016) s.d. ace1988 (6 Desember 2016), halaman *web* tidak bisa  
 1089 dibuka. Hal ini disebabkan karena perbedaan konfigurasi pada *file* "database.php". Pada *commit*  
 1090 f8c64fc (6 Desember 2016) s.d. 57f5ea4 (15 Desember 2016) dan *commit* f1ea410 (5 Januari 2017),  
 1091 di dalam *file* "database.php" terdapat *password*. *Commit* lainnya tidak terdapat *password* pada *file*  
 1092 "database.php". Penulis menggunakan *password* "piktora" pada konfigurasi *database* di MySQL  
 1093 Server.

Listing 3.2: Isi *file* "database.php" pada *commit* f8c64fc(6 Desember 2016)

```

1094 $active_group = 'default';
1095 $query_builder = TRUE;
1096
1097 $db[ 'default' ] = array(
1098     'dsn' => '',
1099     'hostname' => 'localhost',
1100     'username' => 'piktora',
1101     'password' => 'dmHx64%6',
1102     'database' => 'piktora',
1103     'dbdriver' => 'mysqli',
1104     'dbprefix' => '',
1105     'pconnect' => FALSE,

```

```

1103 'db_debug' => (ENVIRONMENT !== 'production') ,
1104 'cache_on' => FALSE,
1105 'cachedir' => '',
1106 'char_set' => 'utf8',
1107 'dbcollat' => 'utf8_general_ci',
1108 'swap_pre' => '',
1109 'encrypt' => FALSE,
1110 'compress' => FALSE,
1111 'stricton' => FALSE,
1112 'failover' => array(),
1113 'save_queries' => TRUE
1114 );

```

Listing 3.2 merupakan isi dari file "database.php" pada commit f8c64fc (6 Desember 2016). Dapat dilihat bahwa *password* yang terdapat pada file "database.php" adalah dmHx64%6. Commit f8c64fc (6 Desember 2016) s.d. ace1988 (6 Desember 2016) menggunakan *password* dmHx64%6, sedangkan commit 627e65b (6 Desember 2016) s.d. 57f5ea4 (15 Desember 2016) dan f1ea410 (5 Januari 2017) menggunakan *password* **piktora**. Karena konfigurasi *password* pada file "database.php" dan phpMyAdmin berbeda, halaman website pada commit f8c64fc (6 Desember 2016) s.d. ace1988 (6 Desember 2016) tidak bisa dibuka.

Sama seperti subbab 3.4.1, solusi untuk mengatasi masalah ini adalah dengan menambahkan *option* yang akan menjalankan *terminal command*. Dimana *terminal command* ini berisi perintah untuk menjalankan *script* PHP. *Script* ini akan mengecek *password* yang terdapat pada file "database.php". Jika tidak ditemukan *password* atau ditemukan *password* berupa **piktora**, maka *script* tidak akan mengubah isi file "database.php". Jika ditemukan *password* berupa dmHx64%6, maka *script* akan mengubah *password* menjadi **piktora**.

### 1131 3.4.3 Permasalahan Migrasi *Database*

1132 Pada commit 3d79d0a (27 Nov 2016), terjadi *error* saat melakukan migrasi database. Pada commit 1133 a19e7f2 (27 November 2016), terdapat satu file untuk melakukan migrasi yaitu "20161122150000\_Structure.php". Pada commit 3d79d0a (27 Nov 2016) terdapat dua file untuk melakukan migrasi 1134 yaitu "20161122150000\_Structure.php" dan "20161122150001\_InitialData.php". Pada commit 1135 a19e7f2(27 November 2016) file "20161122150001\_InitialData.php" dijalankan saat melakukan 1136 migrasi. Versi migrasi *database* menjadi "20161122150000". Pada commit 3d79d0a (27 Nov 2016), 1137 file "20161122150000\_Structure.php" tidak dijalankan karena dianggap sama dengan versi migrasi 1138 *database* saat ini. Hanya file "20161122150001\_InitialData.php" yang dijalankan pada commit 1139 3d79d0a (27 Nov 2016). Isi file "20161122150001\_InitialData.php" pada commit a19e7f2 (27 No- 1140 vember 2016) dan 3d79d0a (27 Nov 2016) berbeda. Hal ini yang menyebabkan terjadinya *error* saat 1141 melakukan migrasi *database*.

1143 Sama seperti subbab 3.4.1, solusi untuk mengatasi masalah ini adalah dengan menambahkan 1144 *option* yang akan menjalankan *terminal command*. Dimana *terminal command* ini berisi perin- 1145 tah untuk menjalankan *script* PHP. *Script* ini akan melakukan dua pekerjaan. Pertama, *script* 1146 akan menghapus *database* **piktora**. Setelah itu *script* akan membuat *database* **piktora**, kemudian 1147 melakukan migrasi *database*.



1148

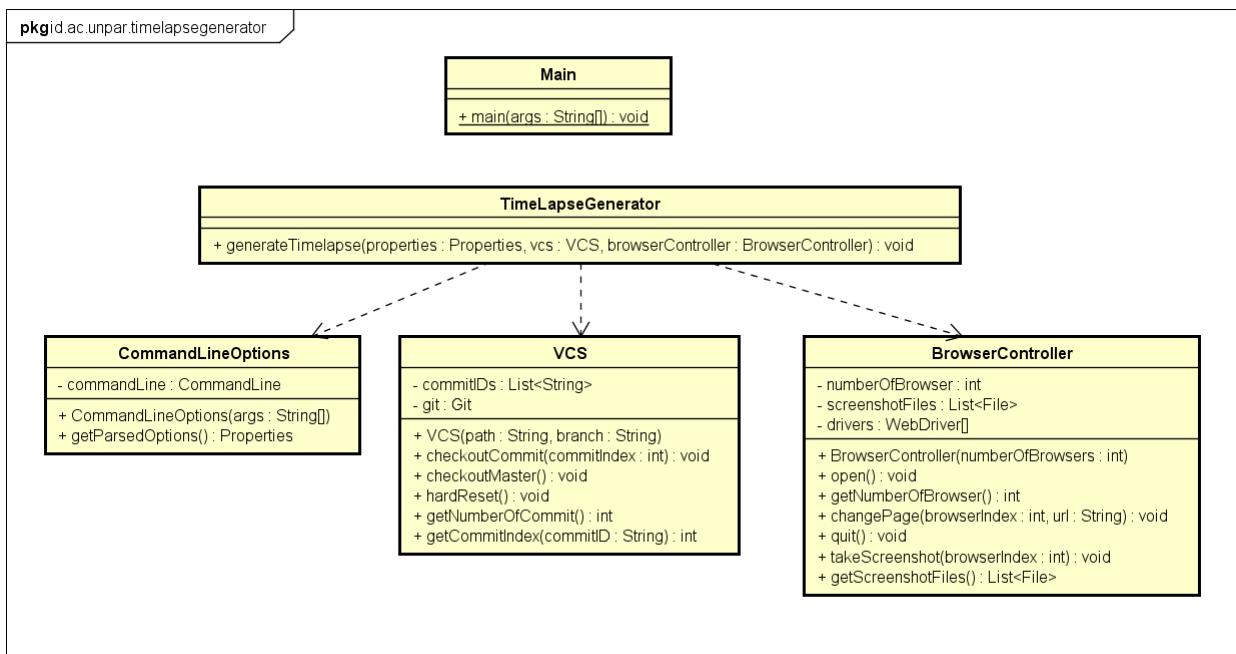
## BAB 4

1149

## PERANCANGAN

1150 Pada bab ini dijelaskan mengenai perancangan perangkat lunak yang dibangun, meliputi peran-  
1151 cangan kelas dan perancangan antarmuka.

1152 **4.1 Perancangan Kelas**



Gambar 4.1: Diagram kelas.

1153 Program pada skripsi ini memiliki lima kelas. Diagram kelas pada program ini dapat dilihat pada  
1154 Gambar 4.1. Berikut adalah rincian kelas yang terdapat pada program ini:

1155 • **BrowserController**

1156 Kelas ini digunakan untuk mengatur *browser*. Operasi-operasi yang dilakukan terhadap  
1157 *browser* yaitu membuka *browser*, mengambil *screenshot*, membuat *browser window* menjadi  
1158 maksimal, dan menutup *browser*. Berikut adalah atribut yang terdapat pada kelas ini:

- 1159 – `private final WebDriver[] drivers`

1160 Atribut ini adalah kumpulan *browser* yang digunakan untuk keperluan *automation testing*.

- private final List<File> screenshotFiles  
Atribut ini berfungsi untuk menyimpan *file* hasil *screenshot*.
- private final int numberOfBrowser  
Atribut ini menyatakan jumlah *browser* yang dimiliki oleh kelas ini. Jumlah maksimal dari *browser* adalah empat.

Berikut adalah *method* yang terdapat pada kelas ini:

- public BrowserController(int numberOfBrowsers)  
Constructor dari kelas ini. Berfungsi untuk menginisialisasi atribut yang dimiliki oleh kelas ini. Parameternya adalah jumlah *browser* yang dapat dimiliki oleh kelas ini.
- public void open()  
Berfungsi untuk membuka semua *browser*, kemudian mengatur ukuran *browser window* menjadi maksimal.
- public int getNumberOfBrowser()  
Berfungsi untuk mengembalikan jumlah browser yang dimiliki kelas ini.
- public void changePage(int browserIndex, String url)  
Berfungsi untuk berpindah halaman pada *browser* tertentu. Parameternya adalah alamat URL untuk berpindah halaman dan indeks *browser* yang akan diubah halamannya.
- public void quit()  
Berfungsi untuk menutup semua *browser*.
- public void takeScreenshot(int browserIndex)  
Berfungsi untuk mengambil *screenshot* pada *browser* tertentu dan menyimpannya ke atribut *screenshotFiles*. Parameternya adalah indeks *browser* yang akan diambil *screenshotnya*.

#### • CommandLineOptions

Kelas ini berfungsi untuk menyimpan semua *option* yang terdapat dalam program ini, dan melakukan *parsing* argumen Command Line Options yang dimasukkan oleh *user*.

Berikut adalah atribut yang terdapat pada kelas ini:

- private final CommandLine commandLine  
Atribut ini berfungsi untuk melakukan *parsing* argumen Command Line Options dan menampung hasilnya.

Berikut adalah *method* yang terdapat pada kelas ini:

- public CommandLineOptions(String[] args)  
Merupakan Constructor dari kelas ini. Berfungsi untuk menentukan *option* yang terdapat pada program dan melakukan parsing argumen Command Line. Parameternya adalah argumen Command Line Option yang didapatkan dari kelas Main.
- public Properties getParsedOptions()  
Berfungsi untuk mengembalikan Command Line Option yang sudah diparsing.

1199 • VCS

1200 Kelas ini digunakan untuk berinteraksi pada proyek perangkat lunak yang terekam oleh Git.  
1201 Berikut adalah atribut yang terdapat pada kelas ini:

- 1202 – private final Git git

1203 Atribut ini digunakan untuk melakukan interaksi pada proyek perangkat lunak yang  
1204 terekam oleh Git.

- 1205 – private final List<String> commitIDs

1206 Atribut ini digunakan untuk menampung seluruh *commit* ID dari hasil penelusuran  
1207 histori.

1208 Berikut ini adalah *method* yang terdapat dalam kelas ini:

- 1209 – public VCS(String path, String branch)

1210 Constructor dari kelas ini. Berfungsi untuk menginisialisasi variabel git dan mendapatkan  
1211 seluruh histori commit proyek perangkat lunak berbasis web pada branch tertentu.  
1212 Dimana branch tersebut diambil dari parameter constructor. Parameternya adalah *path*  
1213 dari proyek perangkat lunak berbasis web dan nama *branch* yang digunakan untuk  
1214 membangkitkan animasi.

- 1215 – public void checkoutCommit(int commitIndex)

1216 Berfungsi untuk melakukan *checkout* ke *commit* tertentu. Parameter dari *method* ini  
1217 adalah indeks dari variabel commitIDs.

- 1218 – public void checkoutMaster()

1219 Berfungsi untuk melakukan *checkout* ke *commit* terakhir.

- 1220 – public void hardReset()

1221 Berfungsi untuk melakukan operasi Git Reset dengan tipe *hard reset*. Operasi ini  
1222 menghapus perubahan pada *working tree* di *commit* tertentu.

- 1223 – public int getNumberOfCommit()

1224 Berfungsi untuk mendapatkan jumlah *commit*.

- 1225 – public int getCommitIndex(String commitID)

1226 Berfungsi untuk mendapatkan indeks dari variabel commitIDs. Parameternya adalah  
1227 *Commit* ID yang akan dicari indeksnya.

1228 • TimeLapseGenerator

1229 Kelas ini digunakan untuk membangkitkan animasi *timelapse*. Berikut adalah *method* yang  
1230 dimiliki oleh kelas ini:

- 1231 – public void generateTimelapse(Properties properties, VCS vcs, BrowserController bro-  
1232 wserController)

1233 Berfungsi untuk membangkitkan animasi *timelapse* berdasarkan langkah-langkah pada  
1234 subbab 3.3. Hasil animasi berupa *file* bertipe GIF. Parameternya adalah objek yang  
1235 bertipe VCS, BrowserController, dan Properties. Parameter properties menampung key  
1236 dan value dari *option* yang sudah diparsing.

## 1237 4.2 Perancangan Antarmuka

1238 Program dalam skripsi ini menggunakan dengan terminal sebagai antarmuka, dengan kata lain  
 1239 menggunakan Command Line Interface. *Input* dari program ini dimasukkan melalui argumen  
 1240 Command Line. *Option* yang dapat dimasukkan ke program ini dapat dilihat pada subbab 3.3.  
 1241 *Output* dari program ini berupa status pada terminal dan *file* hasil animasi bertipe GIF.

Listing 4.1: Status pesan yang akan muncul pada terminal saat program berhasil membangkitkan animasi *timelapse*.

1242 | Animasi timelapse berhasil dibuat

Listing 4.2: Status pesan yang akan muncul pada terminal saat program gagal membangkitkan animasi *timelapse*.

1243 | Animasi timelapse gagal dibuat  
 1244 | <ERROR MESSAGE>

1245 Listing 4.1 menunjukkan status yang akan ditampilkan pada terminal saat program berhasil  
 1246 membangkitkan animasi *timelapse*. Listing 4.2 menunjukkan status yang akan ditampilkan pada  
 1247 terminal saat program gagal membangkitkan animasi *timelapse*. Pada Listing 4.2, baris pertama  
 1248 menunjukkan bahwa program gagal membangkitkan animasi *timelapse*. Pada Listing 4.2, baris  
 1249 kedua menyatakan *error message* dari program. Berikut ini adalah *error message* yang akan  
 1250 ditampilkan saat *user* memasukkan *input* yang tidak valid:

- 1251 • Path proyek tidak valid

1252 Pesan ini muncul jika *path* pada argumen *option project-path* tidak valid.

- 1253 • Jumlah url yang akan dicapture maksimal 4

1254 Pesan ini muncul jika jumlah argumen pada *option capture-url* lebih dari 4.

- 1255 • Seconds per commit harus lebih besar dari 0

1256 Pesan ini muncul jika argumen dari *seconds-per-commit* bernilai lebih kecil dari 0.

- 1257 • Seconds per commit harus kurang dari sama dengan 655

1258 Pesan ini muncul jika argumen dari *seconds-per-commit* bernilai lebih besar dari 655.

- 1259 • Seconds per commit harus berupa bilangan riil atau bilangan bulat

1260 Pesan ini muncul jika argumen dari *seconds-per-commit* bukan bertipe bilangan riil atau  
 1261 bilangan bulat.

- 1262 • Path gambar tidak valid

1263 Pesan ini muncul jika *path* gambar pada argumen *option logo* tidak ditemukan atau *path*  
 1264 gambar bukan merupakan *file* gambar.

- 1265 • Panjang commit ID awal harus berada di antara 7-40 karakter

1266 Pesan ini muncul jika panjang Commit ID pada argumen *option start-commit* kurang dari  
 1267 tujuh karakter atau lebih besar dari empat sepuluh karakter.

- 1268 • Panjang commit ID akhir harus berada di antara 7-40 karakter

1269 Pesan ini muncul jika panjang Commit ID pada argumen *option stop-commit* kurang dari  
1270 tujuh karakter atau lebih besar dari empat sepuluh karakter.

- 1271 • Commit ID awal tidak ditemukan

1272 Pesan ini muncul jika Commit ID pada argumen *option start-commit* tidak ditemukan.

- 1273 • Commit ID akhir tidak ditemukan

1274 Pesan ini muncul jika Commit ID pada argumen *option stop-commit* tidak ditemukan.

- 1275 • Commit ID awal dan akhir tidak boleh sama

1276 Pesan ini muncul jika argumen pada *option start-commit* dan *option stop-commit* bernilai  
1277 sama.

- 1278 • Commit ID awal dan akhir terbalik

1279 Pesan ini muncul jika nilai argumen pada *option start-commit* dan *option stop-commit*  
1280 tertukar.

- 1281 • Branch tidak valid

1282 Pesan ini muncul jika nama branch pada argumen *option branch* tidak valid.

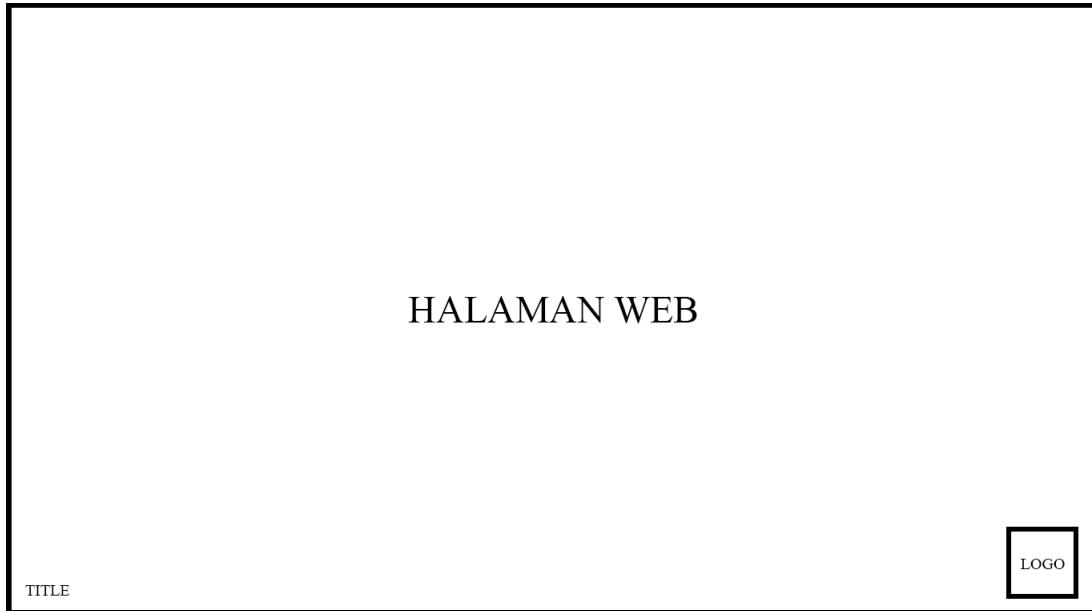
1283 Rancangan *output* dari hasil *file animasi timelapse* dapat dilihat pada Gambar 4.2 sampai dengan

1284 Gambar 4.5. Gambar 4.2 menunjukkan rancangan *output* jika terdapat satu halaman *web*. Gambar

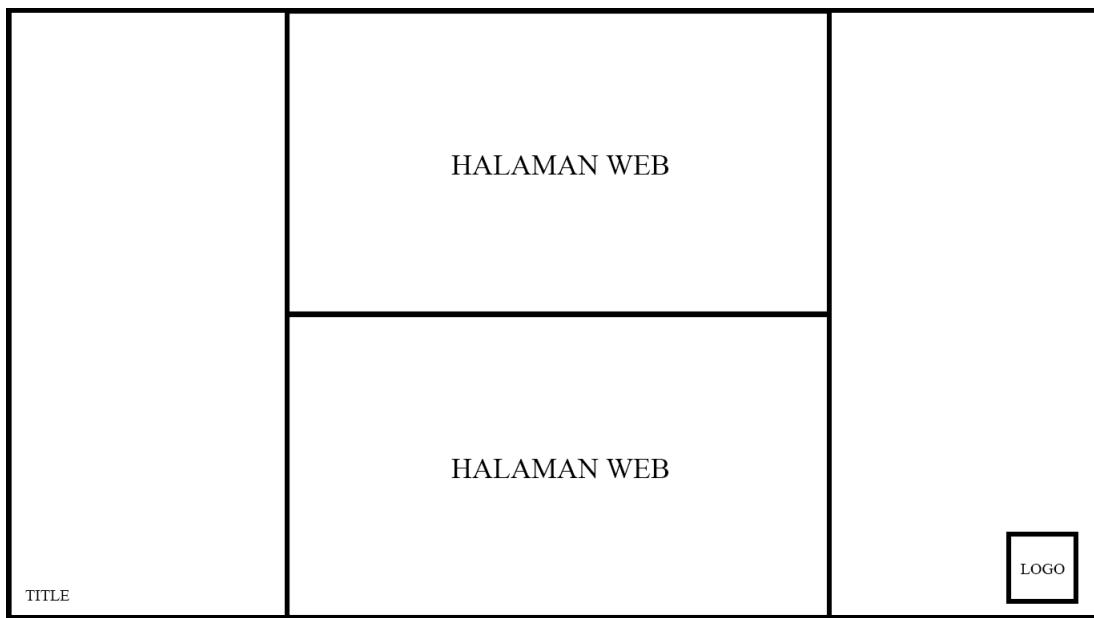
1285 4.3 menunjukkan rancangan *output* jika terdapat dua halaman *web*. Gambar 4.4 menunjukkan

1286 rancangan *output* jika terdapat tiga halaman *web*. Gambar 4.5 menunjukkan rancangan *output* jika

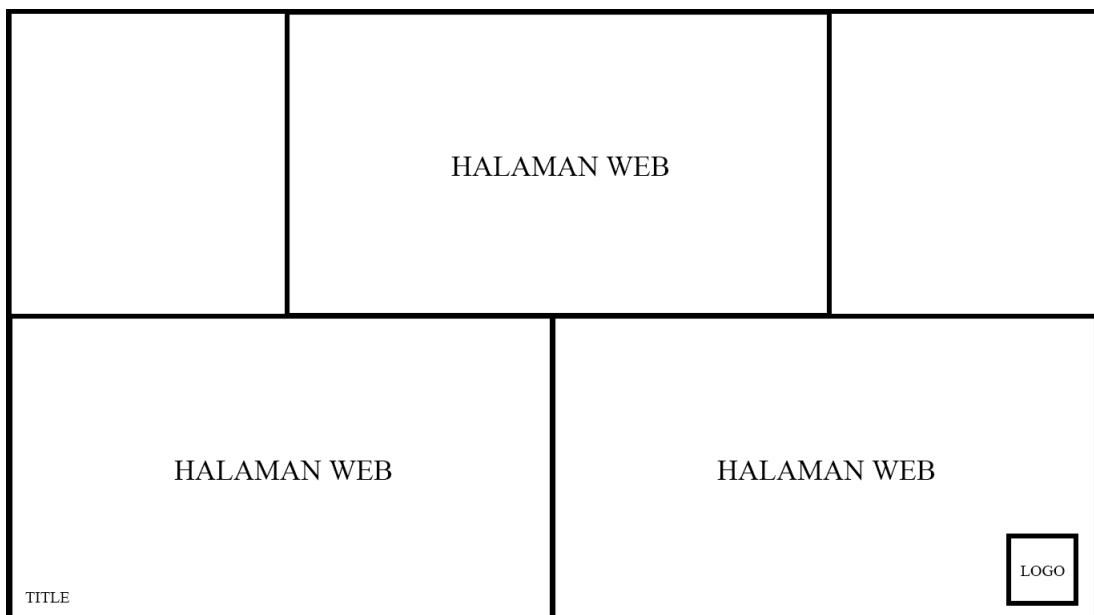
1287 terdapat empat halaman *web*.



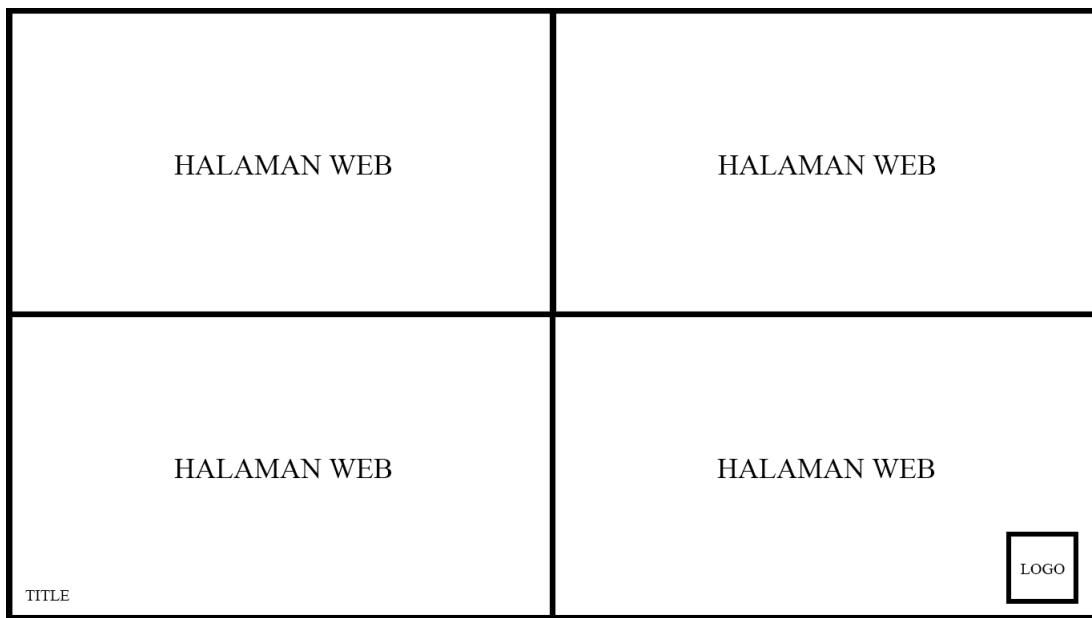
Gambar 4.2: Rancangan *output* jika terdapat satu halaman *web*.



Gambar 4.3: Rancangan *output* jika terdapat dua halaman *web*.



Gambar 4.4: Rancangan *output* jika terdapat tiga halaman *web*.



Gambar 4.5: Rancangan *output* jika terdapat empat halaman *web*.



1288

## BAB 5

1289

### IMPLEMENTASI DAN PENGUJIAN

1290 Pada bab dijelaskan mengenai implementasi perangkat lunak dan pengujian perangkat lunak.  
1291 Bagian implementasi berisi tentang lingkungan implementasi dan hasil implementasi. Bagian  
1292 pengujian berisi tentang pengujian fungsional dan pengujian eksperimental.

1293 **5.1 Implementasi**

1294 **5.1.1 Lingkungan Implementasi**

1295 Implementasi dari perangkat lunak dilakukan pada sebuah laptop. Berikut adalah spesifikasi laptop  
1296 dan perangkat lunak yang digunakan untuk prapengujian:

- 1297 • Processor: Intel Core i3 4030U
- 1298 • RAM: 6GB
- 1299 • Sistem Operasi: Windows 10 pro 64-bit
- 1300 • Versi Apache HTTP Server: 2.4.29
- 1301 • Versi MySQL Server: 5.5.5
- 1302 • Versi Netbeans: 8.1
- 1303 • Versi Google Chrome: 73.0.3683.86

1304 **5.1.2 Hasil Implementasi**

1305 Hasil dari implementasi adalah sebuah perangkat berbasis terminal yang dapat membangkitkan  
1306 animasi *timelapse* pada pengembangan proyek perangkat lunak berbasis *web*. Kode program  
1307 dari perangkat lunak dapat dilihat pada Lampiran A. Setelah dijalankan, perangkat lunak akan  
1308 menghasilkan dua *output* yaitu, status pada terminal dan *file* hasil animasi bertipe GIF.

1309 **1. Status pada Terminal**

1310 Setelah berhasil membangkitkan animasi *timelapse*, perangkat lunak menampilkan status pada  
1311 terminal seperti yang diperlihatkan pada Listing 5.1. Baris 5 menunjukkan bahwa animasi  
1312 *timelapse* berhasil dibangkitkan. Pesan pada baris 1-4 muncul saat ChromeDriver membuka  
1313 dan mulai mengontrol Chrome *browser*.

Listing 5.1: Status pesan pada terminal saat program berhasil membangkitkan animasi *timelapse*.

```
1314 1 Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on
1315   port 16446
1316 2 Only local connections are allowed.
1317 3 Feb 24, 2019 3:26:25 PM org.openqa.selenium.remote.ProtocolHandshake
1318   createSession
1319 4 INFO: Detected dialect: OSS
1320 5 Animasi timelapse berhasil dibuat
```

## 1321 2. **File GIF Hasil Animasi**

1322 Selain menghasilkan status pada terminal, program juga akan menghasilkan sebuah *file* GIF  
1323 hasil animasi. Gambar 5.1 menunjukkan *screenshot* setiap *commit* yang terdapat pada *file*  
1324 GIF hasil animasi dari proyek Piktora. Piktora memiliki 58 *commit*, sehingga terdapat 58  
1325 *screenshot*.



Gambar 5.1: *Screenshot* proyek Piktora pada *commit* 315d374 (31 Oktober 2016) - *commit* 89000be (12 Januari 2018).

## 1326 5.2 Pengujian

### 1327 5.2.1 Pengujian Fungsional

1328 Pengujian ini dilakukan dengan tujuan untuk mengetahui apakah Command Line Option yang  
1329 terdapat pada program sudah berjalan dengan baik. *Option* yang terdapat pada program dapat  
1330 dilihat pada subbab 3.3. Lingkungan pengujian fungsional sama dengan lingkungan implementasi

yang terdapat pada subbab 5.1.1. Pengujian dilakukan pada proyek Piktora dengan menggunakan berbagai macam tes kasus. Histori *commit* proyek Piktora dapat dilihat pada subbab 3.4. Hasil pengujian fungsional dapat dilihat pada Tabel 5.1 sampai dengan Tabel 5.3.

1334

1335

1336



Gambar 5.2: Salah satu *commit* yang terdapat pada file hasil animasi. Terdapat judul dibagian pojok kiri bawah.



Gambar 5.3: Salah satu *commit* yang terdapat pada file hasil animasi. Terdapat logo dibagian pojok kanan bawah.



Gambar 5.4: Salah satu *commit* pada file hasil animasi jika terdapat satu argumen `-capture-url`.

Tabel 5.1: Tabel pengujian fungsional

| No | Tes Kasus  | Output yang diharapkan   | Output program |
|----|--|--|----------------|
| 1. | Menguji <i>option -project-path</i> menggunakan path yang valid. Argumen Command Line yang diberikan ke program:<br><code>-project-path<br/>C:/xampp/htdocs/Piktora/.git</code>  | Program berhasil membangkitkan animasi berdasarkan path yang diberikan dan mengeluarkan status pesan "Animasi timelapse berhasil dibuat" | sesuai         |
| 2. | Menguji <i>option -project-path</i> menggunakan path yang tidak valid. Argumen Command Line yang diberikan ke program:<br><code>-project-path<br/>C:/xampp/htdocs</code>   | Program gagal membuat animasi dan mengeluarkan pesan error "Path proyek tidak valid"   | sesuai         |
| 3. | Menguji <i>option -capture-url</i> menggunakan satu argumen. Argumen Command Line yang diberikan ke program:<br><code>-capture-url<br/>http://localhost</code>   | Program mengambil screenshot berdasarkan alamat yang diberikan. Tampilan dari file hasil animasi dapat dilihat pada Gambar 5.4           | sesuai         |
| 4. | Menguji <i>option -capture-url</i> menggunakan dua argumen. Argumen Command Line yang diberikan ke program:<br><code>-capture-url<br/>http://localhost<br/>http://localhost/about</code>   | Program mengambil screenshot berdasarkan alamat-alamat yang diberikan. Tampilan dari file hasil animasi dapat dilihat pada Gambar 5.5    | sesuai         |
| 5. | Menguji <i>option -capture-url</i> menggunakan tiga argumen. Argumen Command Line yang diberikan ke program:<br><code>-capture-url<br/>http://localhost<br/>http://localhost/about<br/>http://localhost/projects</code>  | Program mengambil screenshot berdasarkan alamat-alamat yang diberikan. Tampilan dari file hasil animasi dapat dilihat pada Gambar 5.6    | sesuai         |
| 6. | Menguji <i>option -capture-url</i> menggunakan empat argumen. Argumen Command Line yang diberikan ke program:<br><code>-capture-url<br/>http://localhost<br/>http://localhost/about<br/>http://localhost/projects<br/>http://localhost/contact</code>                                | Program mengambil screenshot berdasarkan alamat-alamat yang diberikan. Tampilan dari file hasil animasi dapat dilihat pada Gambar 5.7    | sesuai         |
| 7. | Menguji <i>option -capture-url</i> menggunakan lima argumen. Argumen Command Line yang diberikan ke program:<br><code>-capture-url<br/>http://localhost<br/>http://localhost/about<br/>http://localhost/projects<br/>http://localhost/contact<br/>http://localhost/projects/1</code> | Program gagal membuat animasi dan mengeluarkan pesan error "Jumlah url yang akan dicapture maksimal 4"                                   | sesuai         |

Tabel 5.2: Tabel pengujian fungsional

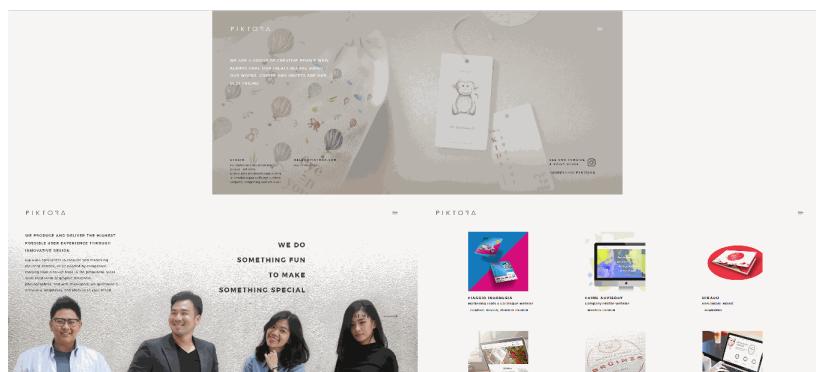
| No  | Tes Kasus  | Output yang diharapkan  | Output program |
|-----|--|---|----------------|
| 8.  | Menguji <i>option -seconds-per-commit</i> dengan nilai 2. Argumen Command Line yang diberikan ke program: <b>-seconds-per-commit 2</b>                                   | Program berhasil membangkitkan animasi dan menghasilkan <i>file</i> GIF yang memiliki durasi 116 detik  | sesuai         |
| 9.  | Menguji <i>option -seconds-per-commit</i> dengan nilai 0. Argumen Command Line yang diberikan ke program: <b>-seconds-per-commit 0</b>                                   | Program gagal membuat animasi dan mengeluarkan pesan error "Seconds per commit harus lebih besar dari 0"  | sesuai         |
| 10. | Menguji <i>option -seconds-per-commit</i> dengan nilai 656. Argumen Command Line yang diberikan ke program: <b>-seconds-per-commit 656</b>                               | Program gagal membuat animasi dan mengeluarkan pesan error "Seconds per commit harus kurang dari sama dengan 655"   | sesuai         |
| 11. | Menguji <i>option -before-capture</i> . Argumen Command Line yang diberikan ke program: <b>-before-capture "php script_piktora.php"</b>                                  | Program menjalankan <i>terminal command</i> sebelum mengambil <i>screenshot</i>   | sesuai         |
| 12. | Menguji <i>option -title</i> . Argumen Command Line yang diberikan ke program: <b>-title Piktora</b>   | Program berhasil membangkitkan animasi dan menghasilkan <i>file</i> GIF, dimana di dalam <i>file</i> tersebut terdapat judul yang terletak di pojok kiri bawah(lihat Gambar 5.2). | sesuai         |
| 13. | Menguji <i>option -logo</i> menggunakan <i>path</i> gambar yang valid. Argumen Command Line yang diberikan ke program: <b>-logo Logo-UNPAR.png</b>                       | Program berhasil membangkitkan animasi dan menghasilkan <i>file</i> GIF, dimana di dalam <i>file</i> tersebut logo yang terletak di pojok kanan bawah (lihat Gambar 5.3)          | sesuai         |
| 14. | Menguji <i>option -logo</i> menggunakan <i>path</i> gambar yang tidak valid. Argumen Command Line yang diberikan ke program: <b>-logo script_piktora.php</b>             | Program gagal membuat animasi dan mengeluarkan pesan error "Path gambar tidak valid"  | sesuai         |
| 15. | Menguji <i>option -branch</i> menggunakan <i>branch</i> yang valid. Argumen Command Line yang diberikan ke program: <b>-branch master</b>                                | Program membangkitkan animasi pada <i>branch</i> master   | sesuai         |
| 16. | Menguji <i>option -branch</i> menggunakan <i>branch</i> yang tidak valid. Argumen Command Line yang diberikan ke program: <b>-branch piktora</b>                         | Program gagal membuat animasi dan mengeluarkan pesan error "Branch tidak valid"   | sesuai         |
| 17. | Menguji <i>option -start-commit</i> menggunakan <i>commit ID</i> dengan panjang 7 karakter. Argumen Command Line yang diberikan ke program: <b>-start-commit 9b0a302</b> | Program membangkitkan animasi dimulai dari commit 9b0a302   | sesuai         |

Tabel 5.3: Tabel pengujian fungsional

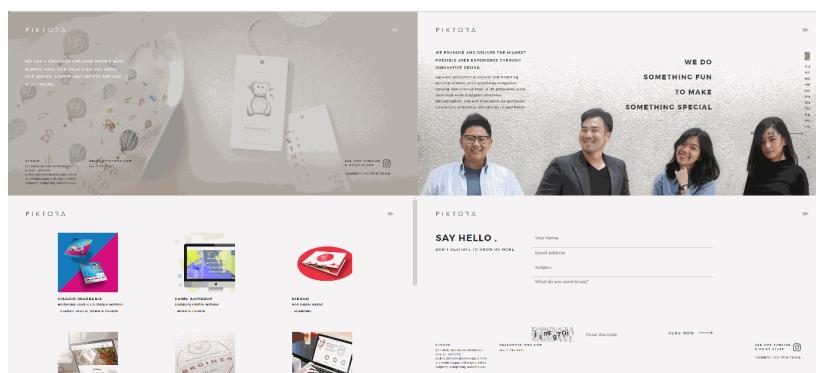
| No  | Tes Kasus  | <i>Output</i> yang diharapkan   | <i>Output</i> program |
|-----|--|---|-----------------------|
| 18. | Menguji <i>option -start-commit</i> menggunakan <i>commit ID</i> dengan panjang kurang dari 7 karakter. Argumen Command Line yang diberikan ke program: <b>-start-commit 9b0a30</b>                                      | Program gagal membuat animasi dan mengeluarkan pesan error "Panjang commit ID awal harus berada di antara 7-40 karakter"  | sesuai                |
| 19. | Menguji <i>option -start-commit</i> menggunakan <i>commit ID</i> dengan panjang lebih dari 40 karakter. Argumen Command Line yang diberikan ke program: <b>-start-commit 9b0a3023ac8c94c67f4c1b39388277768a483dba9</b>   | Program gagal membuat animasi dan mengeluarkan pesan error "Panjang commit ID awal harus berada di antara 7-40 karakter"  | sesuai                |
| 20. | Menguji <i>option -stop-commit</i> menggunakan <i>commit ID</i> dengan panjang 7 karakter. Argumen Command Line yang diberikan ke program: <b>-stop-commit 6a085c1</b>   | Program membangkitkan animasi dimulai dari <i>commit 315d374</i> sampai dengan <i>commit 6a085c1</i>                      | sesuai                |
| 21. | Menguji <i>option -stop-commit</i> menggunakan <i>commit ID</i> dengan panjang kurang dari 7 karakter. Argumen Command Line yang diberikan ke program: <b>-stop-commit 6a085c</b>  | Program gagal membuat animasi dan mengeluarkan pesan error "Panjang commit ID akhir harus berada di antara 7-40 karakter" | sesuai                |
| 22. | Menguji <i>option -stop-commit</i> menggunakan <i>commit ID</i> dengan panjang lebih dari 40 karakter. Argumen Command Line yang diberikan ke program: <b>-stop-commit 6a085c1c37949e6308cfe06a117302e528388e549</b>     | Program gagal membuat animasi dan mengeluarkan pesan error "Panjang commit ID akhir harus berada di antara 7-40 karakter" | sesuai                |
| 23. | Menguji <i>option -start-commit</i> dan <i>option -stop-commit</i> menggunakan rentang <i>commit</i> yang valid. Argumen Command Line yang diberikan ke program: <b>-start-commit 9b0a302 -stop-commit 6a085c1</b>       | Program membangkitkan animasi dimulai dari <i>commit 9b0a302</i> sampai dengan <i>commit 6a085c1</i>                      | sesuai                |
| 24. | Menguji <i>option -start-commit</i> dan <i>option -stop-commit</i> menggunakan rentang <i>commit</i> yang tidak valid. Argumen Command Line yang diberikan ke program: <b>-start-commit 6a085c1 -stop-commit 9b0a302</b> | Program gagal membuat animasi dan mengeluarkan pesan error "Commit ID awal dan akhir terbalik"                            | sesuai                |
| 25. | Menguji <i>option -start-commit</i> dan <i>option -stop-commit</i> menggunakan rentang <i>commit</i> yang tidak valid. Argumen Command Line yang diberikan ke program: <b>-start-commit 9b0a302 -stop-commit 9b0a302</b> | Program gagal membuat animasi dan mengeluarkan pesan error "Commit ID awal dan akhir tidak boleh sama"                    | sesuai                |



Gambar 5.5: Salah satu *commit* pada *file* hasil animasi jika terdapat dua argumen `-capture-url`.



Gambar 5.6: Salah satu *commit* pada *file* hasil animasi jika terdapat tiga argumen `-capture-url`.



Gambar 5.7: Salah satu *commit* pada *file* hasil animasi jika terdapat empat argumen `-capture-url`.

### <sup>1337</sup> 5.2.2 Pengujian Eksperimental

<sup>1338</sup> Pengujian eksperimental ini dibagi menjadi dua bagian. Pengujian eksperimental bagian pertama  
<sup>1339</sup> akan menguji program menggunakan proyek Piktora dengan WebDriver yang berbeda. WebDriver  
<sup>1340</sup> yang digunakan untuk pengujian ini yaitu FirefoxDriver, EdgeDriver, OperaDriver, dan Interne-  
<sup>1341</sup> tExplorerDriver. Pengujian eksperimental bagian kedua akan menguji program dengan situs web

1342 Bootstrap<sup>1</sup>, Netflix Open Source Software Center<sup>2</sup>, IBM Open Source<sup>3</sup>, React<sup>4</sup>, dan Yelp Open  
1343 Source<sup>5</sup>.

1344 Seperti yang sudah dijelaskan pada subbab 2.3 FirefoxDriver, EdgeDriver, OperaDriver, dan  
1345 InternetExplorerDriver merupakan implementasi dari WebDriver yang mengontrol *browser* yang  
1346 mengontrol *browser* tertentu. FirefoxDriver mengontrol Firefox *browser*, OperaDriver mengontrol  
1347 Opera *browser*, dst. Tujuan dari digunakannya berbagai macam WebDriver ini adalah untuk menguji  
1348 program dalam membangkitkan animasi menggunakan berbagai macam *browser*.

1349 Pengujian kedua ini dilakukan dengan tujuan menguji program dengan berbagai situs *web* yang  
1350 berbeda dan memiliki jumlah *commit* yang berbeda. Situs *web* Bootstrap, Netflix Open Source  
1351 Software Center, IBM Open Source, React, dan Yelp Open Source dipilih karena bersifat Open  
1352 Source, repositorinya tersimpan pada GitHub, dan dihosting menggunakan GitHub Pages. GitHub  
1353 Pages<sup>6</sup> merupakan sebuah layanan *hosting* situs *web* untuk melakukan *hosting* halaman *web* secara  
1354 langsung dari repositori GitHub.

1355 Berikut adalah rincian dari pengujian eksperimental:

### 1356 1. Pengujian Proyek Piktora dengan FirefoxDriver

1357 Pengujian pada proyek Piktora dilakukan menggunakan FirefoxDriver. Pada saat melakukan  
1358 pengujian, kode program pada kelas BrowserController baris ke-46 diubah (lihat Lampiran  
1359 A). *Object* bertipe WebDriver dinisialisasi menggunakan *object* bertipe FirefoxDriver. Versi  
1360 Firefox *browser* yang digunakan untuk pengujian adalah 66.0.2. Gambar 5.8, menunjukkan  
1361 tampilan Firefox *browser* saat dikontrol oleh FirefoxDriver. Berikut ini adalah *option* yang  
1362 digunakan untuk menguji program:

- 1363 • -project-path C:/xampp/htdocs/Piktora/.git
- 1364 • -capture-url http://localhost
- 1365 • -before-capture "php script\_piktora.php"

1366 Program berhasil membangkitkan animasi *timelapse* pada proyek Piktora menggunakan  
1367 FirefoxDriver.

---

<sup>1</sup><https://getbootstrap.com/>

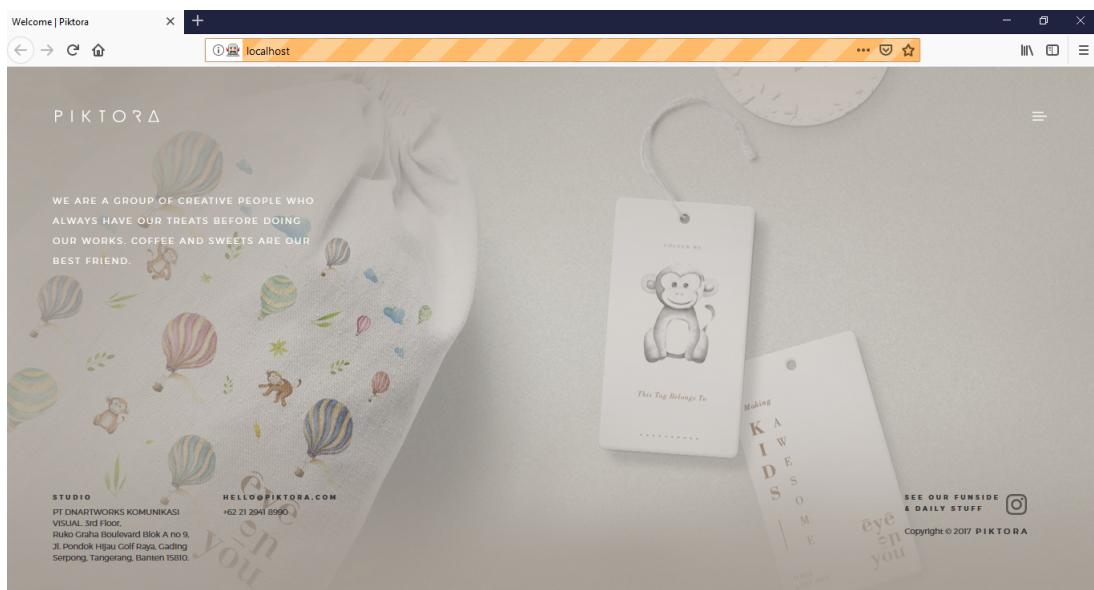
<sup>2</sup><https://netflix.github.io/>

<sup>3</sup><https://ibm.github.io/>

<sup>4</sup><https://reactjs.org/>

<sup>5</sup><https://yelp.github.io/>

<sup>6</sup><https://pages.github.com/>



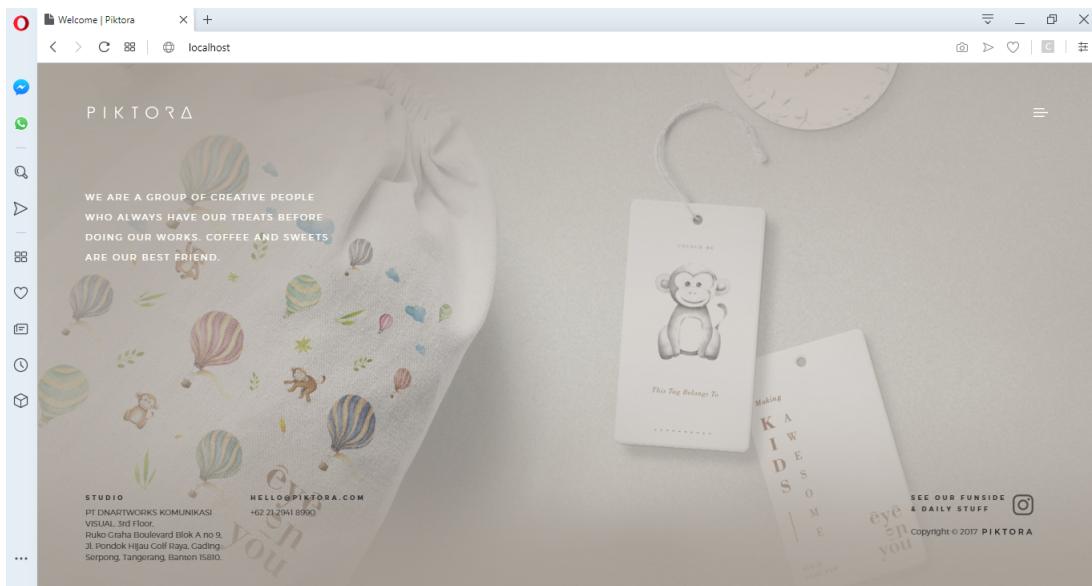
Gambar 5.8: Tampilan pada Firefox *browser* saat dikontrol oleh FirefoxDriver.

## 1368 2. Pengujian Proyek Piktora dengan OperaDriver

1369 Pengujian pada proyek Piktora dilakukan menggunakan OperaDriver. Pada saat melakukan  
 1370 pengujian, kode program pada kelas BrowserController baris ke-46 diubah (lihat Lampiran A).  
 1371 *Object* bertipe WebDriver dinisialisasi menggunakan *object* bertipe OperaDriver. Versi Opera  
 1372 *browser* yang digunakan untuk pengujian adalah 60.0.3255.27 (portable version). Gambar  
 1373 5.9 menunjukkan tampilan pada Opera *browser* saat dikontrol oleh OperaDriver. Berikut ini  
 1374 adalah *option* yang digunakan untuk menguji program:

- 1375 • -project-path C:/xampp/htdocs/Piktora/.git
- 1376 • -capture-url http://localhost
- 1377 • -before-capture "php script\_piktora.php"

1378 Terdapat masalah saat melakukan pengujian. Awalnya Opera *browser* yang digunakan bukan  
 1379 versi *portable* melainkan versi standar. Saat program dijalankan, program mengeluarkan  
 1380 pesan *error* berupa *unknown error: cannot find Opera binary*. Setelah ditelusuri, tidak  
 1381 ditemukan *file* "opera.exe" di dalam direktori "C:/Program Files" atau "C:/Program Files  
 1382 (x86)". Solusi untuk mengatasi masalah ini adalah melakukan instalasi Opera *browser* versi  
 1383 *portable* pada direktori "C:/Program Files" atau "C:/Program Files (x86)". Setelah dilakukan  
 1384 instalasi tersebut, program berhasil membangkitkan animasi *timelapse* pada proyek Piktora  
 1385 menggunakan OperaDriver.



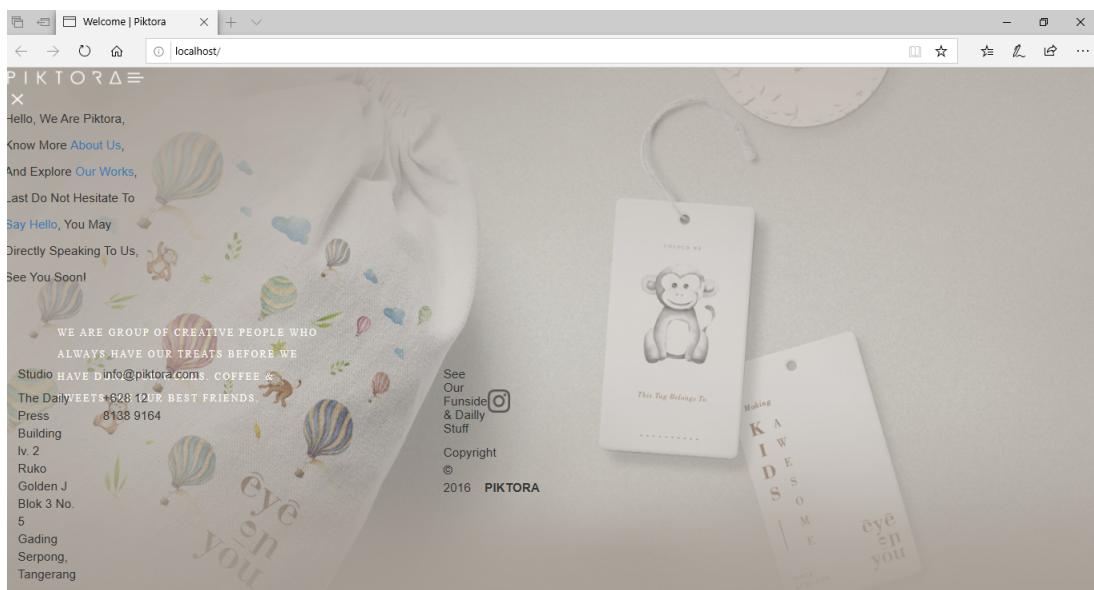
Gambar 5.9: Tampilan pada Opera *browser* saat dikontrol oleh OperaDriver.

### 3. Pengujian Proyek Piktora dengan EdgeDriver

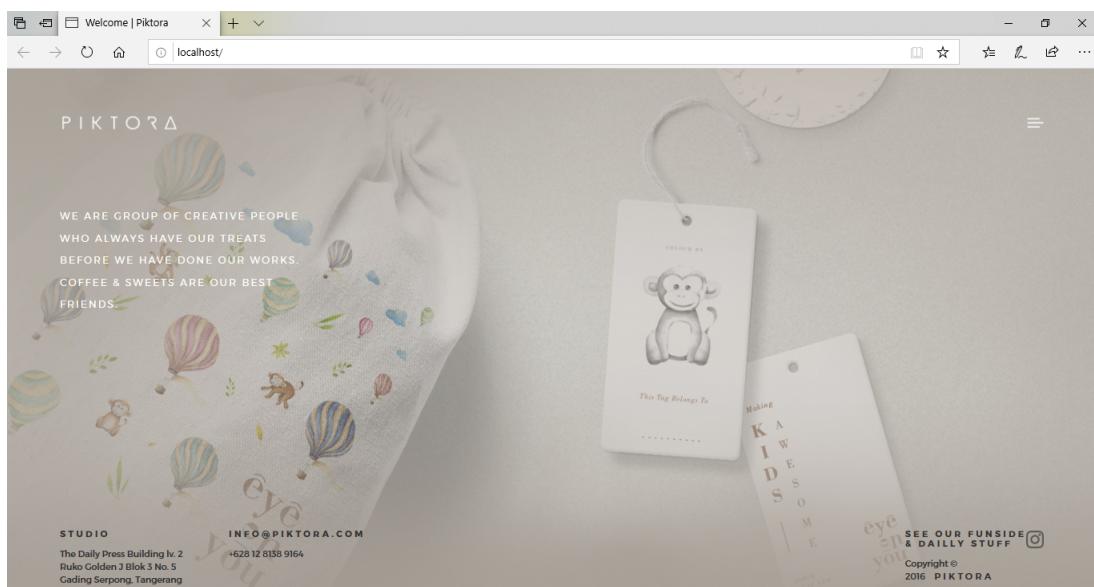
Pengujian pada proyek Piktora dilakukan menggunakan EdgeDriver. Pada saat melakukan pengujian, kode program pada kelas BrowserController baris ke-46 diubah (lihat Lampiran A). *Object* bertipe WebDriver dinisialisasi menggunakan *object* bertipe EdgeDriver. Versi Microsoft Edge *browser* yang digunakan untuk pengujian adalah 44.17763.1.0. Berikut ini adalah *option* yang digunakan untuk menguji program:

- `-project-path C:/xampp/htdocs/Piktora/.git`
- `-capture-url http://localhost`
- `-before-capture "php script_piktora.php"`

Hasil pengujian ini tidak sesuai dengan harapan. Tampilan halaman *web* yang ditampilkan oleh Microsoft Edge *browser* saat dikontrol oleh EdgeDriver tidak rapih seperti yang diperlihatkan pada Gambar 5.10. Tidak diketahui apa yang menyebabkan halaman *web* tersebut menjadi tidak rapih. Sebagai perbandingan, Gambar 5.11 menunjukkan tampilan halaman *web* yang ditampilkan oleh Microsoft Edge *browser* saat tidak dikontrol oleh EdgeDriver. Meskipun tampilan dari halaman *web* tidak rapih, program tetap dapat membangkitkan animasi pada proyek Piktora.



Gambar 5.10: Tampilan halaman *web* pada *browser* saat dikontrol oleh EdgeDriver.



Gambar 5.11: Tampilan halaman *web* pada *browser* saat tidak dikontrol oleh EdgeDriver.

#### 4. Pengujian Proyek Piktora dengan InternetExplorer

Pengujian pada proyek Piktora dilakukan menggunakan InternetExplorerDriver. Pada saat melakukan pengujian, kode program pada kelas BrowserController baris ke-46 diubah (lihat Lampiran A). *Object* bertipe WebDriver dinisialisasi menggunakan *object* bertipe InternetExplorerDriver. Versi Internet Explorer *browser* yang digunakan untuk pengujian adalah 11.379.17763.0. Gambar 5.12 dan Gambar 5.13 menunjukkan tampilan pada Internet Explorer *browser* saat sedang dikontrol oleh InternetExplorerDriver. Berikut ini adalah *option* yang digunakan untuk menguji program:

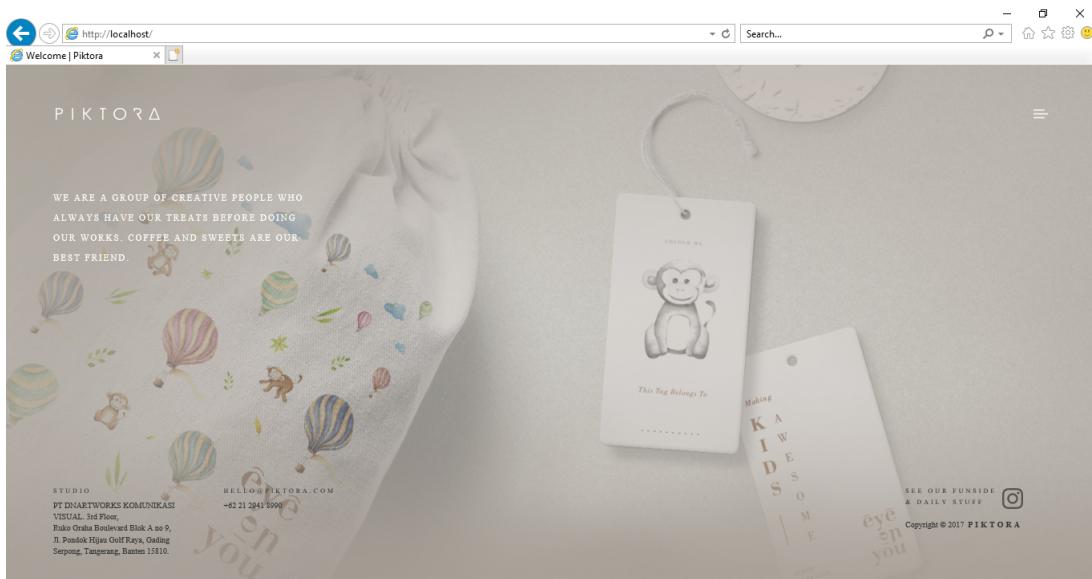
- `-project-path C:/xampp/htdocs/Piktora/.git`
- `-capture-url http://localhost`

- 1412     ● -before-capture "php script\_piktora.php"

1413     Program berhasil membangkitkan animasi *timelapse* pada proyek Piktora menggunakan  
 1414     InternetExplorerDriver.



Gambar 5.12: Keadaan awal dari *browser* saat dikontrol oleh InternetExplorerDriver.



Gambar 5.13: Tampilan pada *browser* saat dikontrol oleh InternetExplorerDriver.

## 5. Pengujian Situs *web* Netflix Open Source Software Center

1415     Netflix Open Source Software Center merupakan proyek Open Source yang dimiliki oleh  
 1416     Netflix. Repotori situs *web* ini disimpan pada GitHub<sup>7</sup>. Repotori ini memiliki 393 *commit*.  
 1417     Lingkungan pengujian eksperimental ini sama dengan lingkungan implementasi yang terdapat  
 1418     pada subbab 5.1.1. Berikut ini adalah *option* yang digunakan untuk menguji program:

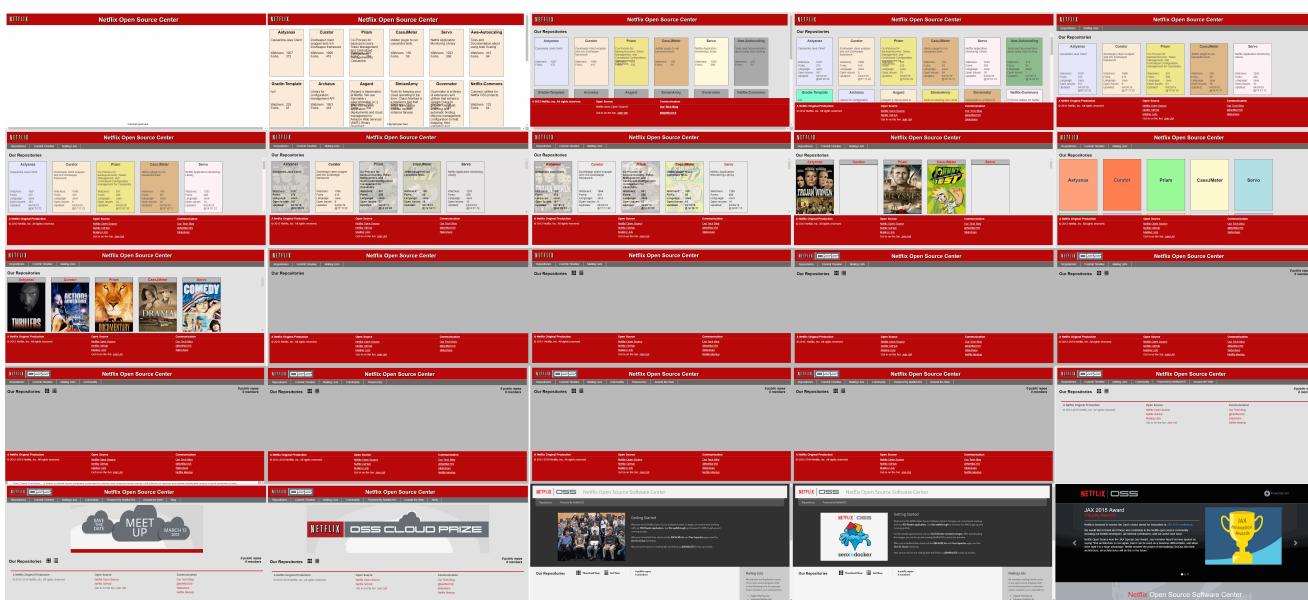
<sup>7</sup><https://github.com/Netflix/netflix.github.com>

- ```

1420 • -project-path C:/xampp/htdocs/netflix.github.com/.git
1421 • -capture-url http://localhost
1422 • -seconds-per-commit 0.1

```

1423 Program berhasil membangkitkan animasi *timelapse* dari situs *web* Netflix Open Source  
1424 Software Center. Tidak ditemukan masalah saat melakukan pengujian. Hasil dari pengujian  
1425 berupa *file* hasil animasi bertipe GIF dengan ukuran 17.5 MB dan berdurasi 39 detik. Sebagian  
1426 dari hasil animasi dapat dilihat pada Gambar 5.14, sedangkan *file* aslinya dapat dilihat pada  
1427 *link* berikut<sup>8</sup>.



Gambar 5.14: Sebagian hasil animasi dari situs *web* Netflix Open Source Software Center.

## 6. Pengujian Situs *web* Bootstrap

1428 Bootstrap adalah kelas *open source* untuk membangun *website* yang dipakai bersama dengan  
1429 HTML, CSS, dan JavaScript. Reposisori situs *web* ini disimpan pada GitHub<sup>9</sup>. Pengujian  
1430 ini dilakukan pada *branch* gh-pages, dimana di pada *branch* tersebut terdapat 8547 *commit*.  
1431 Lingkungan pengujian eksperimental ini sama dengan lingkungan implementasi yang terdapat  
1432 pada subbab 5.1.1. Berikut ini adalah *option* yang digunakan untuk menguji program:

- ```

1434 • -project-path C:/xampp/htdocs/bootstrap/.git
1435 • -capture-url http://localhost
1436 • -seconds-per-commit 0.05
1437 • -branch gh-pages

```

1438 Terdapat masalah saat melakukan pengujian situs *web* Bootstrap. Program suatu ketika  
1439 berhenti dan mengeluarkan pesan error: "short SHA1 685039d is ambiguous". Pesan error ini  
1440 muncul karena terdapat dua Git *object* yang mempunyai 7 digit ID yang sama, sehingga tidak

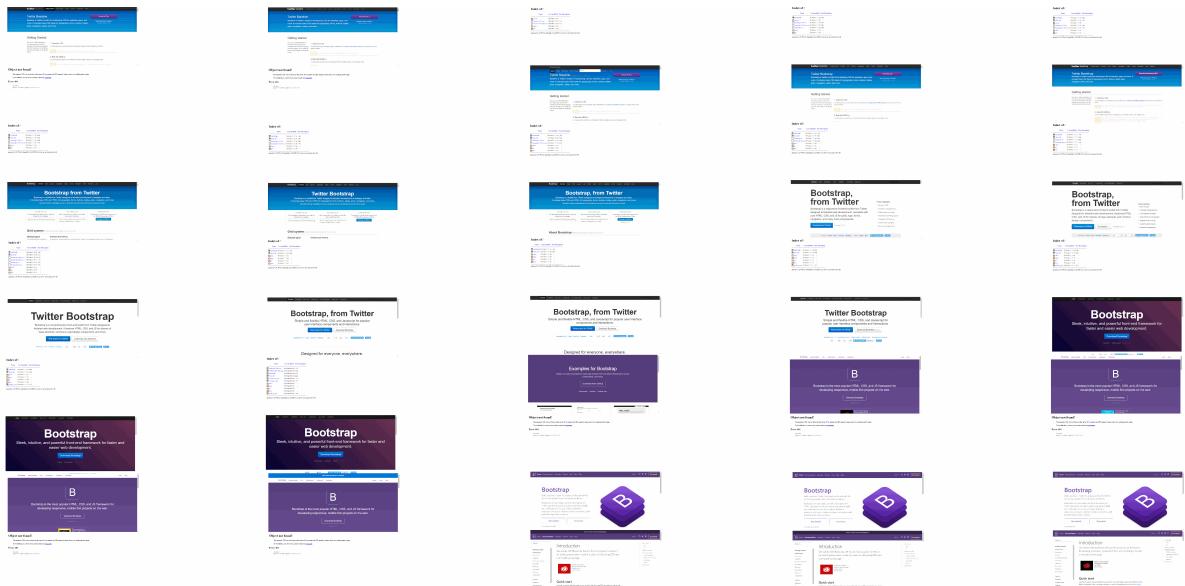
<sup>8</sup><https://github.com/billyAdi/Skripsi/tree/master/Program/TimeLapseGenerator/hasil%20pengujian/Netflix>

<sup>9</sup><https://github.com/twbs/bootstrap>

bisa melakukan *checkout* ke *commit* 685039d. Solusi untuk mengatasi masalah tersebut yaitu dengan mengubah kode program di kelas VCS. Awalnya program hanya menyimpan *commit* ID dengan panjang 7 digit. Setelah itu kode program diubah supaya bisa menyimpan *commit* ID dengan panjang SHA seluruhnya 40 digit.

Pada beberapa *commit*, file "index.html" tidak terdapat pada *root directory*. Pada beberapa *commit*, file "index.html" terletak pada direktori "docs". Karena tidak terdapat file "index.html" halaman web menjadi tidak muncul, yang muncul adalah struktur direktori dari repositori Boostrap.

Setelah itu dilakukan pengujian lagi dengan menggunakan dua argumen pada `-capture-url` saat menjalankan program. Setelah menambahkan argumen berupa `http://localhost/docs` pada option `-capture-url` dan mengubah kode program di kelas VCS, program berhasil membangkitkan animasi. Hasil dari pengujian berupa file hasil animasi bertipe GIF dengan ukuran 160 MB dan berdurasi 7 menit 7 detik. Sebagian dari hasil animasi dapat dilihat pada Gambar 5.15.



Gambar 5.15: Sebagian hasil animasi dari situs web Bootstrap.

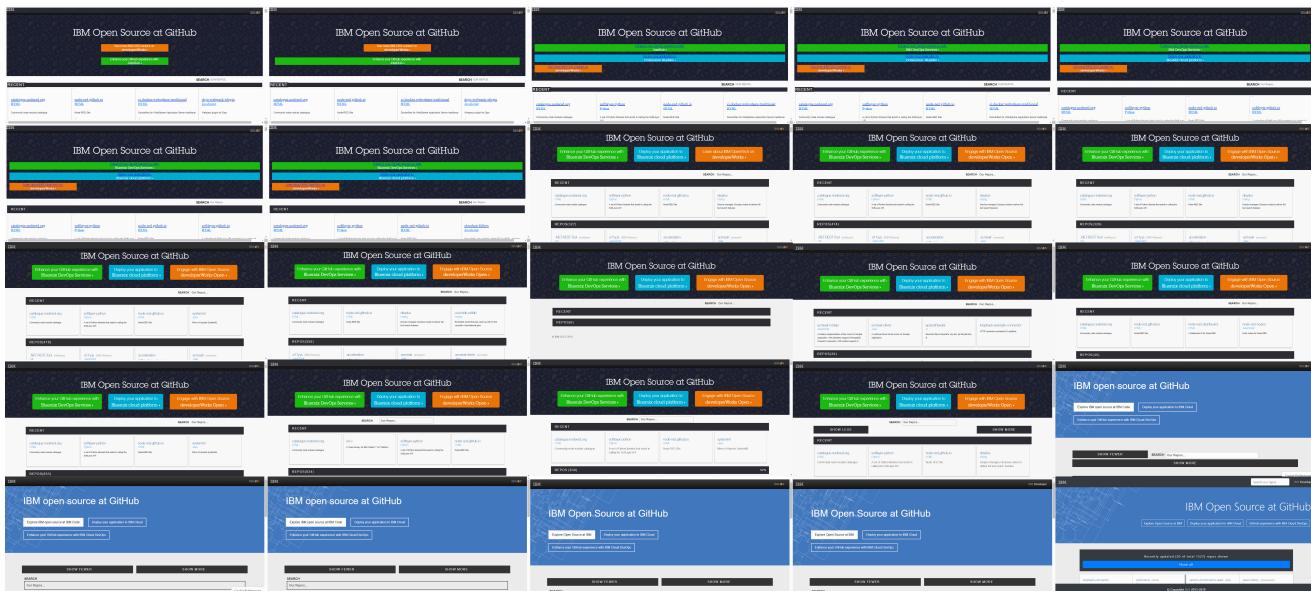
## 7. Pengujian Situs web IBM Open Source

IBM Open Source merupakan proyek Open Source yang dimiliki oleh IBM. Repositori situs web ini disimpan pada GitHub<sup>10</sup>. Repositori ini memiliki 263 *commit*. Lingkungan pengujian eksperimental ini sama dengan lingkungan implementasi yang terdapat pada subbab 5.1.1. Berikut ini adalah *option* yang digunakan untuk menguji program:

- `-project-path C:/xampp/htdocs/ibm.github.io/.git`
- `-capture-url http://localhost`
- `-seconds-per-commit 0.1`

<sup>10</sup><https://github.com/IBM/ibm.github.io>

1463 Program berhasil membangkitkan animasi *timelapse* dari situs *web* IBM Open Source. Tidak  
 1464 ditemukan masalah saat melakukan pengujian. Hasil dari pengujian berupa *file* hasil animasi  
 1465 bertipe GIF dengan ukuran 15.2 MB dan berdurasi 26 detik. Sebagian dari hasil animasi  
 1466 dapat dilihat pada Gambar 5.16, sedangkan *file* aslinya dapat dilihat pada *link* berikut<sup>11</sup>.



Gambar 5.16: Sebagian hasil animasi dari situs *web* IBM Open Source.

## 8. Pengujian Situs *web* React

1468 React merupakan *library* Java Script yang digunakan untuk membangun antarmuka. Repon-  
 1469 sitori situs *web* ini disimpan pada GitHub<sup>12</sup>. Repotori ini memiliki 570 *commit*. Lingkungan  
 1470 pengujian eksperimental ini sama dengan lingkungan implementasi yang terdapat pada subbab  
 1471 5.1.1. Berikut ini adalah *option* yang digunakan untuk menguji program:

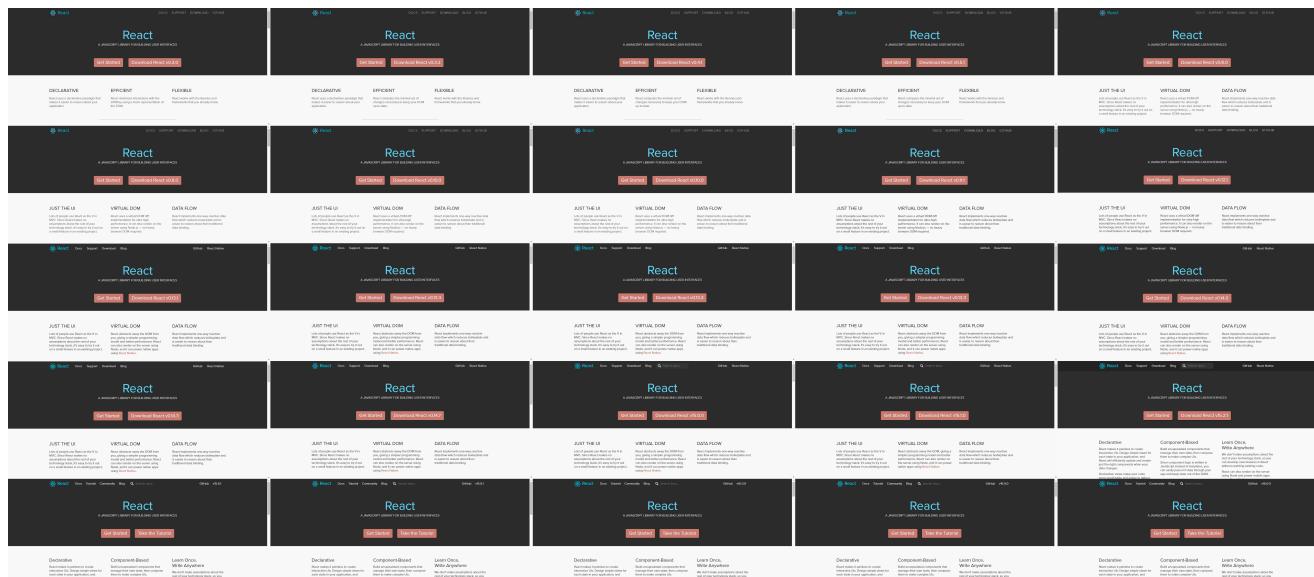
- 1472 • -project-path C:/xampp/htdocs/react/.git
- 1473 • -capture-url http://localhost
- 1474 • -seconds-per-commit 0.05
- 1475 • -branch gh-pages

1476 Program berhasil membangkitkan animasi *timelapse* dari situs *web* React. Untuk beberapa  
 1477 *commit* tidak terdapat *file* "index.html". Hasil dari pengujian berupa *file* hasil animasi bertipe  
 1478 GIF dengan ukuran 18.2 MB dan berdurasi 28 detik. Sebagian dari hasil animasi dapat dilihat  
 1479 pada Gambar 5.17, sedangkan *file* aslinya dapat dilihat pada *link* berikut<sup>13</sup>.

<sup>11</sup><https://github.com/billyAdi/Skripsi/tree/master/Program/TimeLapseGenerator/hasil%20pengujian/IBM>

<sup>12</sup><https://github.com/facebook/react>

<sup>13</sup><https://github.com/billyAdi/Skripsi/tree/master/Program/TimeLapseGenerator/hasil%20pengujian/React>



Gambar 5.17: Sebagian hasil animasi dari situs web React.

## 9. Pengujian Situs web Yelp Open Source

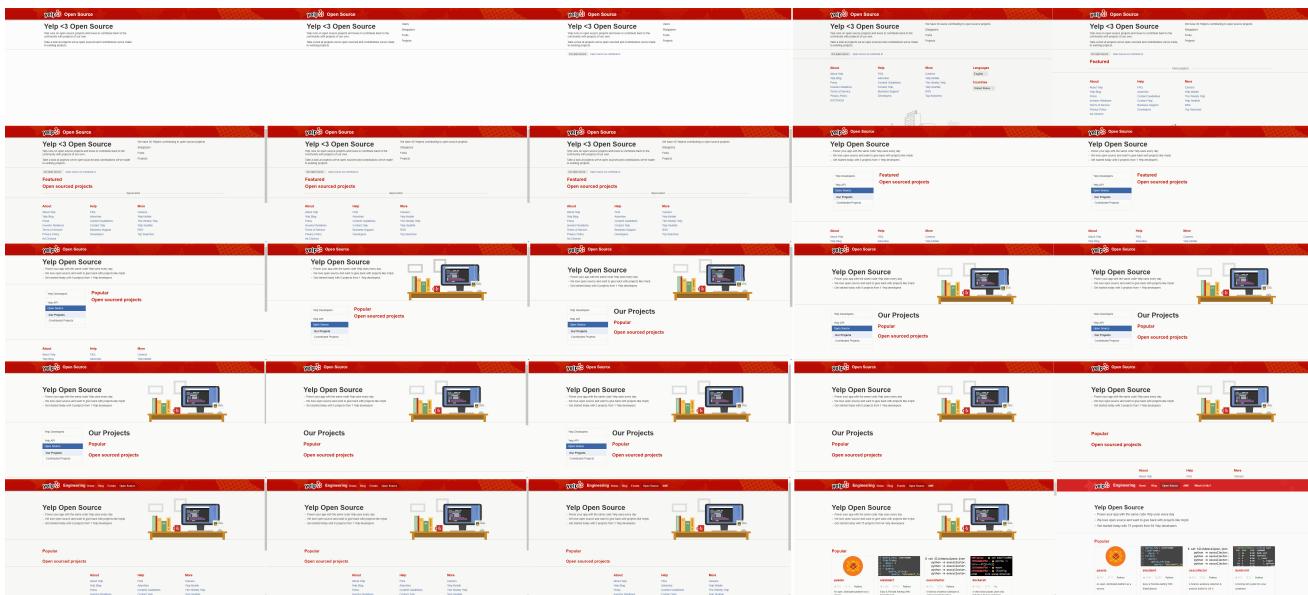
Yelp Open Source merupakan proyek Open Source yang dimiliki oleh Yelp. Reposisori situs web ini disimpan pada GitHub<sup>14</sup>. Reposisori ini memiliki 99 commit. Lingkungan pengujian eksperimental ini sama dengan lingkungan implementasi yang terdapat pada subbab 5.1.1. Berikut ini adalah *option* yang digunakan untuk menguji program:

- `-project-path C:/xampp/htdocs/yelp.github.io/.git`
- `-capture-url http://localhost`
- `-seconds-per-commit 0.2`

Program berhasil membangkitkan animasi *timelapse* dari situs web Yelp. Tidak ditemukan masalah saat melakukan pengujian. Hasil dari pengujian berupa file hasil animasi bertipe GIF dengan ukuran 4.63 MB dan berdurasi 19 detik. Sebagian dari hasil animasi dapat dilihat pada Gambar 5.18 , sedangkan file aslinya dapat dilihat pada link berikut<sup>15</sup>.

<sup>14</sup><https://github.com/Yelp/yelp.github.io>

<sup>15</sup><https://github.com/billyAdi/Skripsi/tree/master/Program/TimeLapseGenerator/hasil%20pengujian/Yelp>



Gambar 5.18: Sebagian hasil animasi dari situs *web* Yelp.

Tabel 5.4: Tabel hasil pengujian eksperimental menggunakan beberapa situs *web*

| No | Situs <i>web</i>                    | Jumlah <i>commit</i> | Ukuran <i>output file</i> | Durasi animasi  |
|----|-------------------------------------|----------------------|---------------------------|-----------------|
| 1. | Yelp Open Source                    | 99                   | 4.63 MB                   | 19 detik        |
| 2. | IBM Open Source                     | 263                  | 15.2 MB                   | 26 detik        |
| 3. | Netflix Open Source Software Center | 393                  | 17.5 MB                   | 39 detik        |
| 4. | React                               | 570                  | 18.2 MB                   | 28 detik        |
| 5. | Bootstrap                           | 8547                 | 160 MB                    | 7 menit 7 detik |

1492 Tabel 5.4 merupakan rekap dari hasil pengujian eksperimental menggunakan beberapa situs *web*.  
 1493 Pada Tabel 5.4 dapat dilihat hubungan antara ukuran *output file* dengan jumlah *commit*. Dapat  
 1494 dilihat bahwa semakin besar jumlah *commit*, ukuran *file* yang dihasilkan semakin besar.

1495 Hal yang menarik dari pengujian eksperimental adalah letak *file* "index.html" yang tidak konsisten  
 1496 pada repositori situs *web* Bootstrap. Pada beberapa *commit*, *file* "index.html" berada pada *root*  
 1497 *directory*. Pada beberapa *commit*, *file* "index.html" berada pada direktori "docs". Pada beberapa  
 1498 *commit* tidak terdapat *file* "index.html". Pada beberapa *commit* terakhir, saat membuka *file*  
 1499 "index.html" pada direktori "docs", halaman menjadi dialihkan ke situs *web* Bootstrap<sup>16</sup>.

1500 Berdasarkan hasil pengujian eksperimental, didapatkan kesimpulan sebagai berikut:

- 1501 1. Semakin besar jumlah *commit* pada repository semakin besar ukuran *file* yang dihasilkan.
- 1502 2. Untuk repositori situs *web* yang sederhana seperti Netflix Open Source Center, Bootstrap,  
 1503 IBM Open Source, React, dan Yelp Open Source, tidak dibutuhkan *option -before-capture*  
 1504 untuk membangkitkan animasi. Untuk repositori yang memerlukan setup basis data seperti  
 1505 Piktora, dibutuhkan *option -before-capture* untuk membangkitkan animasi.

<sup>16</sup><https://getbootstrap.com/docs/4.3/getting-started/introduction>

- 1506        3. Program dapat membangkitkan animasi dengan baik saat menggunakan OperaDriver, Firefo-  
1507        xDriver, dan InternetExplorerDriver. Saat dilakukan pengujian menggunakan EdgeDriver,  
1508        tampilan dari halaman *web* menjadi tidak rapih.



## BAB 6

### KESIMPULAN DAN SARAN

#### 1511 6.1 Kesimpulan

1512 Dari hasil penelitian yang dilakukan, didapatkan kesimpulan-kesimpulan sebagai berikut:

- 1513 • Animasi *timelapse* pada pengembangan proyek perangkat lunak berbasis *web* dapat dibangkitkan dengan bantuan Git. Dengan Git, bisa didapatkan halaman-halaman *web* dari proyek perangkat lunak berbasis web, kemudian *screenshot* dari halaman-halaman tersebut digabung menjadi satu *file* bertipe GIF.
- 1517 • Aplikasi untuk membangkitkan *timelapse* pada pengembangan proyek perangkat lunak berbasis web dapat diimplementasi dengan bantuan *library* JGit, Selenium WebDriver, dan Apache Commons CLI.
- 1520 • Program sudah berjalan dengan baik dan dapat membangkitkan animasi *timelapse* dari proyek pengembangan perangkat lunak berbasis *web*. Hal ini ditunjukkan dengan keberhasilan program dalam membangkitkan animasi pada beberapa situs *web* saat dilakukan pengujian eksperimental.

#### 1524 6.2 Saran

1525 Dari hasil penelitian yang dilakukan, berikut adalah beberapa saran untuk pengembangan:

- 1526 1. Saat ini *output* dari program berupa *file* bertipe GIF. *File* bertipe GIF ini memiliki dua kekurangan, yaitu keterbatasan warna dan ukuran *file* yang besar. GIF memakai sistem palet warna dan hanya terbatas pada 256 warna saja. Selain itu, *file* hasil animasi dari situs *web* Bootstrap berukuran 160 MB. Dalam pengembangan berikutnya, sebaiknya menggunakan format *file* yang lebih modern seperti MP4 atau format lainnya.
- 1531 2. Saat ini ukuran screenshot dari halaman *web* bergantung pada resolusi layar. Pada beberapa situs *web*, tinggi dari halaman *web* melebihi tinggi dari resolusi layar sehingga *screenshot* hanya menampilkan sebagian konten dari halaman *web*. Dalam pengembangan berikutnya, sebaiknya program bisa mendapatkan *screenshot* halaman *web* secara keseluruhan.



## DAFTAR REFERENSI

- [1] Chacon, S. dan Straub, B. (2014) *Pro Git* The expert's voice. Apress.
- [2] Jgit | the eclipse foundation. <https://www.eclipse.org/jgit/>. [Online; diakses 2-September-2018].
- [3] Selenium webdriver. <https://www.seleniumhq.org/about/>. [Online; diakses 2-September-2018].
- [4] Jgit - parent 5.0.3.201809091024-r api. <http://download.eclipse.org/jgit/site/5.0.3.201809091024-r/apidocs/index.html>. [Online; diakses 17-September-2018].
- [5] Selenium documentation. <https://www.seleniumhq.org/docs/>. [Online; diakses 17-September-2018].
- [6] Generated documentation. <https://seleniumhq.github.io/selenium/docs/api/java/>. [Online; diakses 17-September-2018].
- [7] Commons - home. <https://commons.apache.org/proper/commons-cli/index.html>. [Online; diakses 11-Oktober-2018].
- [8] Apache commons cli 1.3.1 api. <https://commons.apache.org/proper/commons-cli/javadocs/api-release/index.html>. [Online; diakses 11-Oktober-2018].
- [9] Gource - a software version control visualization tool. <https://gource.io/>. [Online; diakses 29-Oktober-2018].



## LAMPIRAN A

### KODE PROGRAM

Listing A.1: BrowserController.java

```
1 package id.ac.unpar.timelapsegenerator;
2
3 import java.io.File;
4 import java.util.ArrayList;
5 import java.util.List;
6 import org.openqa.selenium.OutputType;
7 import org.openqa.selenium.TakesScreenshot;
8 import org.openqa.selenium.WebDriver;
9 import org.openqa.selenium.chrome.ChromeDriver;
10
11 /**
12 * Kelas ini digunakan untuk mengatur browser. Operasi-operasi yang dilakukan
13 * terhadap browser yaitu membuka browser, mengambil screenshot, membuat window
14 * browser menjadi maksimal, dan menutup browser.
15 *
16 * Hasil screenshot disimpan dalam bentuk List of File. Variabel driver
17 * merupakan suatu representasi dari browser. Kelas ini dapat memiliki maksimal
18 * empat browser.
19 *
20 * @author Billy Adiwijaya
21 */
22 public class BrowserController {
23
24     private final WebDriver[] drivers;
25     private final List<File> screenshotFiles;
26     private final int numberOfWorkers;
27
28     /**
29      * Constructor yang berfungsi untuk menginisialisasi variabel yang dimiliki
30      * oleh kelas ini.
31      *
32      * @param numberOfWorkers jumlah browser.
33      */
34     public BrowserController(int numberOfWorkers) {
35         this.drivers = new WebDriver[numberOfWorkers];
36         this.screenshotFiles = new ArrayList<>();
37         this.numberOfWorkers = numberOfWorkers;
38     }
39
40     /**
41      * Method ini berfungsi untuk membuka semua browser, kemudian mengatur
42      * ukuran window browser menjadi maksimal.
43      */
44     public void open() {
45         for (int i = 0; i < numberOfWorkers; i++) {
46             this.drivers[i] = new ChromeDriver();
47             this.drivers[i].manage().window().maximize();
48         }
49     }
50
51     /**
52      * Method ini berfungsi untuk mengembalikan jumlah browser yang dimiliki
53      * kelas ini.
54      *
55      * @return jumlah browser yang dimiliki oleh kelas ini.
56      */
57     public int getNumberOfWorkers() {
58         return numberOfWorkers;
59     }
60
61     /**
62      * Method ini berfungsi untuk berpindah halaman pada browser tertentu.
63      *
64      * @param browserIndex indeks browser yang akan diubah halamannya.
65      * @param url alamat URL untuk berpindah halaman.
66      */
67     public void changePage(int browserIndex, String url) {
68         this.drivers[browserIndex].get(url);
69     }
70
71     /**
72      * Method ini berfungsi untuk menutup semua browser.
73      */
74     public void quit() {
75         for (WebDriver driver : this.drivers) {
```

```

76         driver.quit();
77     }
78 }
79
80 /**
81 * Method ini berfungsi untuk mengambil screenshot pada browser tertentu dan
82 * menyimpannya ke atribut screenshotFiles.
83 *
84 * @param browserIndex indeks browser yang akan diambil screenshotnya.
85 */
86 public void takeScreenshot(int browserIndex) {
87     this.screenshotFiles.add(((TakesScreenshot) this.drivers[browserIndex]).getScreenshotAs(OutputType.FILE));
88 }
89
90 /**
91 * Method ini berfungsi untuk mengembalikan hasil screenshot.
92 *
93 * @return hasil screenshot berupa List of File.
94 */
95 public List<File> getScreenshotFiles() {
96     return this.screenshotFiles;
97 }
98 }

```

Listing A.2: CommandLineOptions.java

```

1 package id.ac.unpar.timelapsegenerator;
2
3 import java.util.Properties;
4 import org.apache.commons.cli.CommandLine;
5 import org.apache.commons.cli.CommandLineParser;
6 import org.apache.commons.cli.DefaultParser;
7 import org.apache.commons.cli.Option;
8 import org.apache.commons.cli.Options;
9 import org.apache.commons.cli.ParseException;
10
11 /**
12 * Kelas ini berfungsi untuk menyimpan semua Option yang terdapat pada program
13 * dan melakukan parsing argumen Command Line Option.
14 *
15 * @author Billy Adiwijaya
16 */
17 public class CommandLineOptions {
18
19     private final CommandLine commandLine;
20
21     /**
22      * Constructor dari kelas ini. Berfungsi untuk menentukan Option yang
23      * terdapat pada program dan melakukan parsing argumen Command Line.
24      *
25      * @param args merupakan argumen Command Line Option yang didapatkan dari
26      * kelas Main.
27      * @throws ParseException jika terjadi masalah saat melakukan parsing atau
28      * jumlah argumen capture-url lebih dari 4.
29      */
30     public CommandLineOptions(String[] args) throws ParseException {
31         CommandLineParser parser = new DefaultParser();
32
33         Options options = new Options();
34         options.addOption(Option.builder().required().hasArgs().longOpt("capture-url").argName("url").desc("link yang akan di
35             capture").build());
36         options.addOption(Option.builder().required().longOpt("project-path").argName("path").hasArg().desc("path proyek perangkat
37             lunak").build());
38         options.addOption(Option.builder().longOpt("seconds-per-commit").argName("seconds").hasArg().desc("durasi satu commit").
39             build());
40         options.addOption(Option.builder().longOpt("before-capture").argName("terminal command").hasArg().desc("terminal command
41             yang dijalankan sebelum melakukan screenshot").build());
42         options.addOption(Option.builder().longOpt("start-commit").argName("commit id").hasArg().desc("commit id awal untuk
43             memangkitkan animasi").build());
44         options.addOption(Option.builder().longOpt("stop-commit").argName("commit id").hasArg().desc("commit id akhir untuk
45             memangkitkan animasi").build());
46         options.addOption(Option.builder().longOpt("title").argName("title").hasArg().desc("judul proyek yang akan ditampilkan di
47             pojok kiri bawah").build());
48         options.addOption(Option.builder().longOpt("logo").argName("image path").hasArg().desc("logo yang akan ditampilkan di
49             pojok kanan bawah").build());
50         options.addOption(Option.builder().longOpt("branch").argName("branch").hasArg().desc("branch yang digunakan untuk
51             membangkitkan animasi").build());
52
53         this.commandLine = parser.parse(options, args);
54         if (this.commandLine.getOptionValues("capture-url").length > 4) {
55             throw new ParseException("Jumlah url yang akan dicapture maksimal 4");
56         }
57
58     /**
59      * Method ini berfungsi untuk mengembalikan Option yang sudah diparsing.
60      *
61      * @return Option yang sudah diparsing berupa objek dengan tipe Properties.
62      */
63     public Properties getParsedOptions() {
64         Properties properties = new Properties();
65         for (Option option : this.commandLine.getOptions()) {
66             if (option.getLongOpt().equals("capture-url")) {
67                 String[] values = option.getValues();
68                 String value = values[0];
69
70                 for (int i = 1; i < values.length; i++) {
71                     value = value + ";" + values[i];
72                 }
73             }
74         }
75     }
76 }

```

---

```

64         }
65         properties.setProperty(option.getLongOpt(), value);
66     } else {
67         properties.setProperty(option.getLongOpt(), option.getValue());
68     }
69 }
70 return properties;
71 }
72 }
```

Listing A.3: Main.java

```

1 package id.ac.unpar.timelapsegenerator;
2
3 import java.io.File;
4 import java.util.Properties;
5 import javax.imageio.ImageIO;
6 import org.eclipse.jgit.api.errors.GitAPIException;
7
8 /**
9  * @author Billy Adiwijaya
10 */
11 public class Main {
12
13     public static void main(String[] args) {
14         CommandLineOptions commandLineOptions = null;
15         try {
16             commandLineOptions = new CommandLineOptions(args);
17         } catch (Exception e) {
18             System.out.println("Animasi timelapse gagal dibuat");
19             System.out.println(e.getMessage());
20             System.exit(0);
21         }
22         Properties properties = commandLineOptions.getParsedOptions();
23         int numberofBrowsers = properties.getProperty("capture-url").split(";").length;
24
25         VCS vcs = null;
26         try {
27             String branch="master";
28             if(properties.getProperty("branch")!=null){
29                 branch=properties.getProperty("branch");
30             }
31             vcs = new VCS(properties.getProperty("project-path"),branch);
32             if (properties.getProperty("seconds-per-commit") != null) {
33                 if (Double.parseDouble(properties.getProperty("seconds-per-commit")) <= 0) {
34                     throw new Exception("Seconds per commit harus lebih besar dari 0");
35                 }
36                 else if(Double.parseDouble(properties.getProperty("seconds-per-commit"))>655){
37                     throw new Exception("Seconds per commit harus kurang dari sama dengan 655");
38                 }
39             }
40
41             if (properties.getProperty("logo") != null) {
42                 File file = new File(properties.getProperty("logo"));
43                 if (!file.exists() || ImageIO.read(new File(properties.getProperty("logo"))) == null) {
44                     throw new Exception("Path gambar tidak valid");
45                 }
46             }
47
48             if (properties.getProperty("start-commit") != null) {
49                 if (properties.getProperty("start-commit").length() < 7 || properties.getProperty("start-commit").length() > 40) {
50                     throw new Exception("Panjang commit ID awal harus berada di antara 7-40 karakter");
51                 }
52                 if (vcs.getCommitIndex(properties.getProperty("start-commit")) == -1) {
53                     throw new Exception("Commit ID awal tidak ditemukan");
54                 }
55             }
56
57             if (properties.getProperty("stop-commit") != null) {
58                 if (properties.getProperty("stop-commit").length() < 7 || properties.getProperty("stop-commit").length() > 40) {
59                     throw new Exception("Panjang commit ID akhir harus berada di antara 7-40 karakter");
60                 }
61                 if (vcs.getCommitIndex(properties.getProperty("stop-commit")) == -1) {
62                     throw new Exception("Commit ID akhir tidak ditemukan");
63                 }
64             }
65
66             if (properties.getProperty("start-commit") != null && properties.getProperty("stop-commit") != null) {
67                 if (vcs.getCommitIndex(properties.getProperty("start-commit")) > vcs.getCommitIndex(properties.getProperty("stop-
68                     commit"))) {
69                     throw new Exception("Commit ID awal dan akhir terbalik");
70                 } else if (vcs.getCommitIndex(properties.getProperty("start-commit")) == vcs.getCommitIndex(properties.getProperty(
71                     "stop-commit"))) {
72                     throw new Exception("Commit ID awal dan akhir tidak boleh sama");
73                 }
74             }
75         } catch (NumberFormatException e) {
76             System.out.println("Animasi timelapse gagal dibuat");
77             System.out.println("Seconds per commit harus berupa bilangan riil atau bilangan bulat");
78             System.exit(0);
79         } catch (Exception e) {
80             System.out.println("Animasi timelapse gagal dibuat");
81             System.out.println(e.getMessage());
82             System.exit(0);
83         }
84     BrowserController browserController = new BrowserController(numberofBrowsers);
```

```

85     TimeLapseGenerator timeLapseGenerator = new TimeLapseGenerator();
86     try {
87         timeLapseGenerator.generateTimelapse(properties, vcs, browserController);
88     } catch (Exception e) {
89         try {
90             vcs.checkoutMaster();
91         } catch (GitAPIException ex) {
92         } finally {
93             System.out.println("Animasi timelapse gagal dibuat");
94             System.out.println(e.getMessage());
95             System.exit(0);
96         }
97     }
98     System.out.println("Animasi timelapse berhasil dibuat");
99 }
100 }
```

Listing A.4: TimeLapseGenerator.java

```

1 package id.ac.unpar.timelapsegenerator;
2
3 import java.awt.Color;
4 import java.awt.Font;
5 import java.awt.FontMetrics;
6 import java.awt.Graphics2D;
7 import java.awt.image	BufferedImage;
8 import java.io.File;
9 import java.io.IOException;
10 import java.text.SimpleDateFormat;
11 import java.util.ArrayList;
12 import java.util.Date;
13 import java.util.List;
14 import java.util.Properties;
15 import javax.imageio.ImageIO;
16 import javax.imageio.stream.FileImageOutputStream;
17 import javax.imageio.stream.ImageOutputStream;
18 import org.apache.commons.io.FileUtils;
19 import org.eclipse.jgit.api.errors.GitAPIException;
20
21 /**
22 * Kelas ini digunakan untuk membangkitkan animasi timelapse.
23 *
24 * @author Billy Adiwijaya
25 */
26 public class TimeLapseGenerator {
27
28     /**
29      * Method ini berfungsi untuk membangkitkan animasi timelapse. Hasil dari
30      * animasi berupa File dengan tipe GIF.
31      *
32      * @param properties variabel yang menampung key dan value Option yang sudah
33      * diparsing.
34      * @param vcs variabel bertipe VCS yang digunakan untuk berinteraksi pada
35      * proyek perangkat lunak berbasis web yang terekam oleh Git.
36      * @param browserController variabel bertipe BrowserController untuk mengatur
37      * browser.
38      * @throws GitAPIException jika terjadi masalah saat melakukan operasi Git
39      * Checkout atau Git Reset.
40      * @throws IOException jika terjadi masalah saat menjalankan terminal
41      * command.
42      * @throws InterruptedException jika terjadi interupsi pada thread saat
43      * menjalankan terminal command.
44      */
45     public void generateTimelapse(Properties properties, VCS vcs, BrowserController browserController) throws GitAPIException,
46             IOException, InterruptedException {
47         int indexAwal = 0;
48         int indexAkhir = vcs.getNumberOfCommit() - 1;
49
50         if (properties.getProperty("start-commit") != null) {
51             indexAwal = vcs.getCommitIndex(properties.getProperty("start-commit"));
52         }
53
54         if (properties.getProperty("stop-commit") != null) {
55             indexAkhir = vcs.getCommitIndex(properties.getProperty("stop-commit"));
56         }
57
58         String captureURL[] = properties.getProperty("capture-url").split(";");
59         browserController.open();
60         for (int i = indexAwal; i <= indexAkhir; i++) {
61             vcs.checkoutCommit(i);
62
63             if (properties.getProperty("before-capture") != null) {
64                 Process process;
65                 process = Runtime.getRuntime().exec(properties.getProperty("before-capture"));
66                 process.waitFor();
67             }
68
69             for (int j = 0; j < captureURL.length; j++) {
70                 browserController.changePage(j, captureURL[j]);
71                 browserController.takeScreenshot(j);
72             }
73
74             vcs.hardReset();
75         }
76         browserController.quit();
77         vcs.checkoutMaster();
78
79         List<File> screenshotFiles = new ArrayList<>();
```

```

79| List<File> resultImages = new ArrayList<>();
80| screenshotFiles = browserController.getScreenshotFiles();
81|
82| switch (browserController.getNumberOfBrowser()) {
83|     case 1:
84|         resultImages = screenshotFiles;
85|         break;
86|     case 2:
87|         for (int i = 0, j = 0; i < screenshotFiles.size() / browserController.getNumberOfBrowser(); i++, j += 2) {
88|             resultImages.add(screenshotFiles.get(i));
89|             BufferedImage bufferedImage1, bufferedImage2, bufferedImageResult;
90|             bufferedImage1 = ImageIO.read(screenshotFiles.get(j));
91|             bufferedImage2 = ImageIO.read(screenshotFiles.get(j + 1));
92|             bufferedImageResult = new BufferedImage(bufferedImage1.getWidth(), bufferedImage1.getHeight(), bufferedImage1.
93|                 getType());
94|
95|             Graphics2D graphics = bufferedImageResult.createGraphics();
96|             graphics.setBackground(Color.WHITE);
97|             graphics.fillRect(0, 0, bufferedImageResult.getWidth(), bufferedImageResult.getHeight());
98|
99|             graphics.drawImage(bufferedImage1, bufferedImage1.getWidth() / 4, 0, bufferedImage1.getWidth() / 2,
100|                 bufferedImage1.getHeight() / 2, null);
101|             graphics.drawImage(bufferedImage2, bufferedImage2.getWidth() / 4, bufferedImage2.getHeight() / 2,
102|                 bufferedImage2.getWidth() / 2, bufferedImage2.getHeight() / 2, null);
103|             graphics.dispose();
104|             ImageIO.write(bufferedImageResult, "png", resultImages.get(i));
105|         }
106|         break;
107|     case 3:
108|         for (int i = 0, j = 0; i < screenshotFiles.size() / browserController.getNumberOfBrowser(); i++, j += 3) {
109|             resultImages.add(screenshotFiles.get(i));
110|             BufferedImage bufferedImage1, bufferedImage2, bufferedImage3, bufferedImageResult;
111|             bufferedImage1 = ImageIO.read(screenshotFiles.get(j));
112|             bufferedImage2 = ImageIO.read(screenshotFiles.get(j + 1));
113|             bufferedImage3 = ImageIO.read(screenshotFiles.get(j + 2));
114|             bufferedImageResult = new BufferedImage(bufferedImage1.getWidth(), bufferedImage1.getHeight(), bufferedImage1.
115|                 getType());
116|
117|             Graphics2D graphics = bufferedImageResult.createGraphics();
118|             graphics.setBackground(Color.WHITE);
119|             graphics.fillRect(0, 0, bufferedImageResult.getWidth(), bufferedImageResult.getHeight());
120|
121|             graphics.drawImage(bufferedImage1, bufferedImage1.getWidth() / 4, 0, bufferedImage1.getWidth() / 2,
122|                 bufferedImage1.getHeight() / 2, null);
123|             graphics.drawImage(bufferedImage2, 0, bufferedImage2.getHeight() / 2, bufferedImage2.getWidth() / 2,
124|                 bufferedImage2.getHeight() / 2, null);
125|             graphics.drawImage(bufferedImage3, bufferedImage3.getWidth() / 2, bufferedImage3.getHeight() / 2,
126|                 bufferedImage3.getWidth() / 2, bufferedImage3.getHeight() / 2, null);
127|             graphics.dispose();
128|             ImageIO.write(bufferedImageResult, "png", resultImages.get(i));
129|         }
130|         break;
131|     case 4:
132|         for (int i = 0, j = 0; i < screenshotFiles.size() / browserController.getNumberOfBrowser(); i++, j += 4) {
133|             resultImages.add(screenshotFiles.get(i));
134|             BufferedImage bufferedImage1, bufferedImage2, bufferedImage3, bufferedImage4, bufferedImageResult;
135|             bufferedImage1 = ImageIO.read(screenshotFiles.get(j));
136|             bufferedImage2 = ImageIO.read(screenshotFiles.get(j + 1));
137|             bufferedImage3 = ImageIO.read(screenshotFiles.get(j + 2));
138|             bufferedImage4 = ImageIO.read(screenshotFiles.get(j + 3));
139|
140|             bufferedImageResult = new BufferedImage(bufferedImage1.getWidth(), bufferedImage1.getHeight(), bufferedImage1.
141|                 getType());
142|
143|             Graphics2D graphics = bufferedImageResult.createGraphics();
144|             graphics.setBackground(Color.WHITE);
145|             graphics.fillRect(0, 0, bufferedImageResult.getWidth(), bufferedImageResult.getHeight());
146|
147|             graphics.drawImage(bufferedImage1, 0, 0, bufferedImage1.getHeight() / 2, bufferedImage1.getHeight() / 2, null);
148|             graphics.drawImage(bufferedImage2, bufferedImage2.getWidth() / 2, 0, bufferedImage2.getWidth() / 2,
149|                 bufferedImage2.getHeight() / 2, null);
150|             graphics.drawImage(bufferedImage3, 0, bufferedImage3.getHeight() / 2, bufferedImage3.getWidth() / 2,
151|                 bufferedImage3.getHeight() / 2, null);
152|             graphics.drawImage(bufferedImage4, bufferedImage4.getWidth() / 2, bufferedImage4.getHeight() / 2,
153|                 bufferedImage4.getWidth() / 2, bufferedImage4.getHeight() / 2, null);
154|             graphics.dispose();
155|             ImageIO.write(bufferedImageResult, "png", resultImages.get(i));
156|         }
157|         break;
158|     default:
159|         break;
160|     }
161|
162|     for (File resultImage : resultImages) {
163|         if (properties.getProperty("title") == null && properties.getProperty("logo") == null) {
164|             break;
165|         } else {
166|             if (properties.getProperty("title") != null) {
167|                 BufferedImage bufferedImage = ImageIO.read(resultImage);
168|                 Graphics2D graphic = (Graphics2D) bufferedImage.getGraphics();
169|                 graphic.setFont(new Font("Times New Roman", Font.BOLD, 18));
170|                 FontMetrics fontMetrics = graphic.getFontMetrics(graphic.getFont());
171|                 graphic.setColor(Color.black);
172|                 graphic.drawString(properties.getProperty("title"), 5, bufferedImage.getHeight() - fontMetrics.getDescent() -
173|                     5);
174|                 graphic.dispose();
175|                 ImageIO.write(bufferedImage, "png", resultImage);
176|             }
177|         }
178|     }
179| }

```

```

166     if (properties.getProperty("logo") != null) {
167         BufferedImage logo = ImageIO.read(new File(properties.getProperty("logo")));
168         BufferedImage bufferedImage = ImageIO.read(resultImage);
169         Graphics2D graphic = (Graphics2D) bufferedImage.getGraphics();
170         graphic.drawImage(logo, bufferedImage.getWidth() - logo.getWidth() - 5, bufferedImage.getHeight() - logo.
171             getHeight() - 5, null);
172         graphic.dispose();
173         ImageIO.write(bufferedImage, "png", resultImage);
174     }
175 }
176 String fileName = String.format("%s.gif", new SimpleDateFormat("yyyy-MM-dd.HH.mm.ss").format(new Date()));
177 try (ImageOutputStream output = new FileOutputStream(new File(fileName))) {
178     int frameDelay = 1000;
179     if (properties.getProperty("seconds-per-commit") != null) {
180         frameDelay = (int) (Double.parseDouble(properties.getProperty("seconds-per-commit")) * 1000);
181         if (frameDelay == 0) {
182             frameDelay = 1000;
183         }
184     }
185     GifSequenceWriter writer = new GifSequenceWriter(output, ImageIO.read(resultImages.get(0)).getType(), frameDelay,
186         false);
187     for (File resultImage : resultImages) {
188         BufferedImage bufferedImage = ImageIO.read(resultImage);
189         writer.writeToSequence(bufferedImage);
190     }
191     writer.close();
192 }
193 }
194 }
```

Listing A.5: VCS.java

```

1 package id.ac.unpar.timelapsegenerator;
2
3 import java.io.IOException;
4 import java.util.ArrayList;
5 import java.util.Collections;
6 import java.util.List;
7 import org.eclipse.jgit.api.Git;
8 import org.eclipse.jgit.api.ListBranchCommand;
9 import org.eclipse.jgit.api.ResetCommand.ResetType;
10 import org.eclipse.jgit.api.errors.GitAPIException;
11 import org.eclipse.jgit.internal.storage.file.FileRepository;
12 import org.eclipse.jgit.lib.Ref;
13 import org.eclipse.jgit.lib.Repository;
14 import org.eclipse.jgit.revwalk.RevCommit;
15
16 /**
17 * Kelas ini berfungsi untuk berinteraksi pada proyek perangkat lunak berbasis
18 * web yang terekam oleh Git.
19 *
20 * @author Billy Adiwijaya
21 */
22 public class VCS {
23
24     private final Git git;
25     private final List<String> commitIDs;
26
27     /**
28      * Constructor dari kelas ini. Berfungsi untuk menginisialisasi variabel git
29      * dan mendapatkan seluruh histori commit proyek perangkat lunak berbasis
30      * web pada branch tertentu. Dimana branch tersebut diambil dari parameter
31      * constructor.
32      *
33      * @param path merupakan path dari proyek perangkat lunak berbasis web.
34      * @param branch nama branch yang digunakan untuk membangkitkan animasi.
35      * @throws IOException jika path proyek tidak valid atau repositori tidak
36      * bisa diakses.
37      * @throws GitAPIException jika terjadi masalah saat melakukan operasi Git
38      * Log.
39      * @throws Exception jika branch tidak valid
40      */
41     public VCS(String path, String branch) throws IOException, GitAPIException, Exception {
42         Repository repository = new FileRepository(path);
43         if (repository.getRef("HEAD") == null) {
44             throw new IOException("Path proyek tidak valid");
45         }
46         this.git = new Git(repository);
47         List<Ref> refs = git.branchList().setListMode(ListBranchCommand.ListMode.ALL).call();
48         int refIdx = -1;
49         for (int i = 0; i < refs.size(); i++) {
50             if (refs.get(i).getName().contains(branch)) {
51                 refIdx = i;
52                 break;
53             }
54         }
55         if (refIdx == -1) {
56             throw new Exception("Branch tidak valid");
57         }
58         git.checkout().setName(refs.get(refIdx).getName()).call();
59         Iterable<RevCommit> commits = git.log().call();
60         this.commitIDs = new ArrayList<>();
61         for (RevCommit commit : commits) {
62             this.commitIDs.add(commit.getName());
63         }
64     }
```

```
65     Collections.reverse(commitIDs);
66 }
67 /**
68 * Berfungsi untuk melakukan checkout ke commit tertentu.
69 *
70 * @param commitIndex indeks dari variabel commitIDs.
71 * @throws GitAPIException jika terjadi masalah saat melakukan operasi Git
72 * Checkout.
73 */
74 public void checkoutCommit(int commitIndex) throws GitAPIException {
75     this.git.checkout().setName(this.commitIDs.get(commitIndex)).call();
76 }
77 /**
78 * Berfungsi untuk melakukan checkout ke commit terakhir.
79 *
80 * @throws GitAPIException jika terjadi masalah saat melakukan operasi Git
81 * Checkout.
82 */
83 public void checkoutMaster() throws GitAPIException {
84     this.git.checkout().setName("master").call();
85 }
86 /**
87 * Berfungsi untuk melakukan operasi Git Reset. Operasi ini menghapus
88 * perubahan pada working tree dan staging area di commit tertentu.
89 *
90 * @throws GitAPIException jika terjadi masalah saat melakukan operasi Git
91 * Reset.
92 */
93 public void hardReset() throws GitAPIException {
94     this.git.reset().setMode(ResetType.HARD).call();
95 }
96 /**
97 * Berfungsi untuk mendapatkan jumlah commit.
98 *
99 * @return jumlah commit.
100 */
101 public int getNumberOfCommit() {
102     return this.commitIDs.size();
103 }
104 /**
105 * Berfungsi untuk mendapatkan index dari variabel commitID.
106 *
107 * @param commitID merupakan Commit ID yang akan dicari indeksnya.
108 * @return indeks dari variabel commitIDs.
109 */
110 public int getCommitIndex(String commitID) {
111     int result = -1;
112     for (int i = 0; i < this.commitIDs.size(); i++) {
113         if (commitID.equals(this.commitIDs.get(i).substring(0, commitID.length()))) {
114             result = i;
115             break;
116         }
117     }
118     return result;
119 }
120 }
```



## LAMPIRAN B

### LIBRARY UNTUK MEMBUAT FILE GIF

Listing B.1: GifSequenceWriter.java

```
1 package id.ac.unpar.timelapsegenerator;
2 //
3 //  GifSequenceWriter.java
4 //
5 //  Created by Elliot Kroo on 2009-04-25.
6 //
7 // This work is licensed under the Creative Commons Attribution 3.0 Unported
8 // License. To view a copy of this license, visit
9 // http://creativecommons.org/licenses/by/3.0/ or send a letter to Creative
10 // Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.
11 //
12 //
13 import javax.imageio.*;
14 import javax.imageio.metadata.*;
15 import javax.imageio.stream.*;
16 import java.awt.image.*;
17 import java.io.*;
18 import java.util.Iterator;
19
20 public class GifSequenceWriter {
21     protected ImageWriter gifWriter;
22     protected ImageWriteParam imageWriteParam;
23     protected IIOMetadata imageMetaData;
24
25     /**
26      * Creates a new GifSequenceWriter
27      *
28      * @param outputStream the ImageOutputStream to be written to
29      * @param imageType one of the imageTypes specified in BufferedImage
30      * @param timeBetweenFramesMS the time between frames in milliseconds
31      * @param loopContinuously wether the gif should loop repeatedly
32      * @throws IOException if no gif ImageWriters are found
33      *
34      * @author Elliot Kroo (elliot[at]kroo[dot]net)
35      */
36     public GifSequenceWriter(
37         ImageOutputStream outputStream,
38         int imageType,
39         int timeBetweenFramesMS,
40         boolean loopContinuously) throws IOException, IOException {
41         // my method to create a writer
42         gifWriter = getWriter();
43         imageWriteParam = gifWriter.getDefaultWriteParam();
44         ImageTypeSpecifier imageTypeSpecifier =
45             ImageTypeSpecifier.createFromBufferedImageType(imageType);
46
47         imageMetaData =
48             gifWriter.getDefaultImageMetadata(imageTypeSpecifier,
49             imageWriteParam);
50
51         String metaFormatName = imageMetaData.getNativeMetadataFormatName();
52
53         IIOMetadataNode root = (IIOMetadataNode)
54             imageMetaData.getAsTree(metaFormatName);
55
56         IIOMetadataNode graphicsControlExtensionNode = getNode(
57             root,
58             "GraphicControlExtension");
59
60         graphicsControlExtensionNode.setAttribute("disposalMethod", "none");
61         graphicsControlExtensionNode.setAttribute("userInputFlag", "FALSE");
62         graphicsControlExtensionNode.setAttribute(
63             "transparentColorFlag",
64             "FALSE");
65         graphicsControlExtensionNode.setAttribute(
66             "delayTime",
67             Integer.toString(timeBetweenFramesMS / 10));
68         graphicsControlExtensionNode.setAttribute(
69             "transparentColorIndex",
70             "0");
71
72         IIOMetadataNode commentsNode = getNode(root, "CommentExtensions");
73         commentsNode.setAttribute("CommentExtension", "Created by MAH");
74
75 }
```

```

76 IIOMetadataNode appExtensionsNode = getNode(
77   root,
78   "ApplicationExtensions");
79
80 IIOMetadataNode child = new IIOMetadataNode("ApplicationExtension");
81
82 child.setAttribute("applicationID", "NETSCAPE");
83 child.setAttribute("authenticationCode", "2.0");
84
85 int loop = loopContinuously ? 0 : 1;
86
87 child.setUserObject(new byte[]{ 0x1, (byte) (loop & 0xFF),
88   ((loop >> 8) & 0xFF)} );
89 appExtensionsNode.appendChild(child);
90
91 imageMetaData.setFromTree(metaFormatName, root);
92
93 gifWriter.setOutput(outputStream);
94
95 gifWriter.prepareWriteSequence(null);
96 }
97
98 public void writeToSequence(RenderedImage img) throws IOException {
99   gifWriter.writeToSequence(
100     new IIOMImage(
101       img,
102       null,
103       imageMetaData),
104     imageWriteParam);
105 }
106
107 /**
108 * Close this GifSequenceWriter object. This does not close the underlying
109 * stream, just finishes off the GIF.
110 */
111 public void close() throws IOException {
112   gifWriter.endWriteSequence();
113 }
114
115 /**
116 * Returns the first available GIF ImageWriter using
117 * ImageIO.getImageWritersBySuffix("gif").
118 *
119 * @return a GIF ImageWriter object
120 * @throws IOException if no GIF image writers are returned
121 */
122 private static ImageWriter getWriter() throws IOException {
123   Iterator<ImageWriter> iter = ImageIO.getImageWritersBySuffix("gif");
124   if(iter.hasNext()) {
125     throw new IOException("No GIF Image Writers Exist");
126   } else {
127     return iter.next();
128   }
129 }
130
131 /**
132 * Returns an existing child node, or creates and returns a new child node (if
133 * the requested node does not exist).
134 *
135 * @param rootNode the <tt>IIOMetadataNode</tt> to search for the child node.
136 * @param nodeName the name of the child node.
137 *
138 * @return the child node, if found or a new node created with the given name.
139 */
140 private static IIOMetadataNode getNode(
141   IIOMetadataNode rootNode,
142   String nodeName) {
143   int nNodes = rootNode.getLength();
144   for (int i = 0; i < nNodes; i++) {
145     if (rootNode.item(i).getNodeName().compareToIgnoreCase(nodeName)
146         == 0) {
147       return((IIOMetadataNode) rootNode.item(i));
148     }
149   }
150   IIOMetadataNode node = new IIOMetadataNode(nodeName);
151   rootNode.appendChild(node);
152   return(node);
153 }
154
155 /**
156 * public GifSequenceWriter(
157 *   BufferedOutputStream outputStream,
158 *   int imageType,
159 *   int timeBetweenFramesMS,
160 *   boolean loopContinuously) {
161 */
162
163
164 public static void main(String[] args) throws Exception {
165   if (args.length > 1) {
166     // grab the output image type from the first image in the sequence
167     BufferedImage firstImage = ImageIO.read(new File(args[0]));
168
169     // create a new BufferedOutputStream with the last argument
170     ImageOutputStream output =
171       new FileImageOutputStream(new File(args[args.length - 1]));
172
173     // create a gif sequence with the type of the first image, 1 second
174     // between frames, which loops continuously

```

```
175|     GifSequenceWriter writer =
176|         new GifSequenceWriter(output, firstImage.getType(), 1, false);
177|
178|     // write out the first image to our sequence...
179|     writer.writeToSequence(firstImage);
180|     for(int i=1; i<args.length-1; i++) {
181|         BufferedImage nextImage = ImageIO.read(new File(args[i]));
182|         writer.writeToSequence(nextImage);
183|     }
184|
185|     writer.close();
186|     output.close();
187| } else {
188|     System.out.println(
189|         "Usage: java GifSequenceWriter [list of gif files] [output file]");
190| }
191|
192| }
```



# LAMPIRAN C

## KODE PROGRAM PROYEK PIKTORA PADA COMMIT 89000BE (12 JANUARI 2018)

Listing C.1: auth.php

```
1 | <?php
2 | defined('BASEPATH') OR exit('No direct script access allowed');
3 |
4 | $config['domain'] = 'http://localhost';
5 | $config['google-clientid'] = '9911963967-e4apbt2ksb0492p59rtoc8qhcent6a4e.apps.googleusercontent.com';
6 | $config['google-clientsecret'] = 'tw-7N1uNuqdCPRkXRFVqz7xR';
7 | $config['google-redirecturi'] = $config['domain'] . '/admin/oauth2callback';
8 |
```

Listing C.2: config.php

```
1 | <?php
2 | defined('BASEPATH') OR exit('No direct script access allowed');
3 |
4 | /*
5 | -----
6 | Base Site URL
7 | -----
8 |
9 | URL to your CodeIgniter root. Typically this will be your base URL,
10 | WITH a trailing slash:
11 |
12 | http://example.com/
13 |
14 | WARNING: You MUST set this value!
15 |
16 | If it is not set, then CodeIgniter will try guess the protocol and path
17 | your installation, but due to security concerns the hostname will be set
18 | to $_SERVER['SERVER_ADDR'] if available, or localhost otherwise.
19 | The auto-detection mechanism exists only for convenience during
20 | development and MUST NOT be used in production!
21 |
22 | If you need to allow multiple domains, remember that this file is still
23 | a PHP script and you can easily do that on your own.
24 |
25 */
26 $config['base_url'] = 'http://piktora.localhost';
27
28 /*
29 | -----
30 | Index File
31 | -----
32 |
33 | Typically this will be your index.php file, unless you've renamed it to
34 | something else. If you are using mod_rewrite to remove the page set this
35 | variable so that it is blank.
36 |
37 */
38 $config['index_page'] = '';
39
40 /*
41 | -----
42 | URI PROTOCOL
43 | -----
44 |
45 | This item determines which server global should be used to retrieve the
46 | URI string. The default setting of 'REQUEST_URI' works for most servers.
47 | If your links do not seem to work, try one of the other delicious flavors:
48 |
49 | 'REQUEST_URI'    Uses $_SERVER['REQUEST_URI']
50 | 'QUERY_STRING'   Uses $_SERVER['QUERY_STRING']
51 | 'PATH_INFO'      Uses $_SERVER['PATH_INFO']
52 |
53 | WARNING: If you set this to 'PATH_INFO', URIs will always be URL-decoded!
54 */
55 $config['uri_protocol'] = 'REQUEST_URI';
56
57 /*
58 | -----
59 | URL suffix
60 | -----
```

```
61 | This option allows you to add a suffix to all URLs generated by CodeIgniter.  
62 | For more information please see the user guide:  
63 |  
64 | https://codeigniter.com/user_guide/general/urls.html  
65 */  
66 $config['url_suffix'] = '';  
67 /*  
68 |-----  
69 | Default Language  
70 |-----  
71 |  
72 | This determines which set of language files should be used. Make sure  
73 | there is an available translation if you intend to use something other  
74 | than english.  
75 |  
76 |  
77 |  
78 */  
79 $config['language'] = 'english';  
80 /*  
81 |-----  
82 | Default Character Set  
83 |-----  
84 |  
85 | This determines which character set is used by default in various methods  
86 | that require a character set to be provided.  
87 |  
88 | See http://php.net/htmlspecialchars for a list of supported charsets.  
89 |  
90 |  
91 */  
92 $config['charset'] = 'UTF-8';  
93 /*  
94 |-----  
95 | Enable/Disable System Hooks  
96 |-----  
97 |  
98 | If you would like to use the 'hooks' feature you must enable it by  
99 | setting this variable to TRUE (boolean). See the user guide for details.  
100 |  
101 |  
102 */  
103 $config['enable_hooks'] = FALSE;  
104 /*  
105 |-----  
106 | Class Extension Prefix  
107 |-----  
108 |  
109 |  
110 | This item allows you to set the filename/classname prefix when extending  
111 | native libraries. For more information please see the user guide:  
112 |  
113 | https://codeigniter.com/user_guide/general/core_classes.html  
114 | https://codeigniter.com/user_guide/general/creating_libraries.html  
115 |  
116 */  
117 $config['subclass_prefix'] = 'MY_';  
118 /*  
119 |-----  
120 | Composer auto-loading  
121 |-----  
122 |  
123 | Enabling this setting will tell CodeIgniter to look for a Composer  
124 | package auto-loader script in application/vendor/autoload.php.  
125 |  
126 | $config['composer_autoload'] = TRUE;  
127 |  
128 | Or if you have your vendor/ directory located somewhere else, you  
129 | can opt to set a specific path as well:  
130 |  
131 | $config['composer_autoload'] = '/path/to/vendor/autoload.php';  
132 |  
133 | For more information about Composer, please visit http://getcomposer.org/  
134 |  
135 | Note: This will NOT disable or override the CodeIgniter-specific  
136 | auto-loading (application/config/autoload.php)  
137 |  
138 */  
139 $config['composer_autoload'] = '../vendor/autoload.php';  
140 /*  
141 |-----  
142 | Allowed URL Characters  
143 |-----  
144 |  
145 |  
146 | This lets you specify which characters are permitted within your URLs.  
147 | When someone tries to submit a URL with disallowed characters they will  
148 | get a warning message.  
149 |  
150 | As a security measure you are STRONGLY encouraged to restrict URLs to  
151 | as few characters as possible. By default only these are allowed: a-z 0-9~%.:_-  
152 |  
153 | Leave blank to allow all characters -- but only if you are insane.  
154 |  
155 | The configured value is actually a regular expression character group  
156 | and it will be executed as: ! preg_match('/^<permitted_uri_chars>+$/i  
157 |  
158 | DO NOT CHANGE THIS UNLESS YOU FULLY UNDERSTAND THE REPERCUSSIONS!!  
159 |
```

```

160 */
161 $config['permitted_uri_chars'] = 'a-z 0-9%.:_-';
162 /*
163 |-----
164 | Enable Query Strings
165 |-----
166 |
167 | By default CodeIgniter uses search-engine friendly segment based URLs:
168 | example.com/who/what/where/
169 |
170 | By default CodeIgniter enables access to the $_GET array. If for some
171 | reason you would like to disable it, set 'allow_get_array' to FALSE.
172 |
173 | You can optionally enable standard query string based URLs:
174 | example.com?who=me&what=something&where=here
175 |
176 | Options are: TRUE or FALSE (boolean)
177 |
178 | The other items let you set the query string 'words' that will
179 | invoke your controllers and its functions:
180 | example.com/index.php?c=controller&m=function
181 |
182 | Please note that some of the helpers won't work as expected when
183 | this feature is enabled, since CodeIgniter is designed primarily to
184 | use segment based URLs.
185 |
186 */
187 $config['allow_get_array'] = TRUE;
188 $config['enable_query_strings'] = FALSE;
189 $config['controller_trigger'] = 'c';
190 $config['function_trigger'] = 'm';
191 $config['directory_trigger'] = 'd';
192 |
193 /*
194 |-----
195 | Error Logging Threshold
196 |-----
197 |
198 | You can enable error logging by setting a threshold over zero. The
199 | threshold determines what gets logged. Threshold options are:
200 |
201 | 0 = Disables logging, Error logging TURNED OFF
202 | 1 = Error Messages (including PHP errors)
203 | 2 = Debug Messages
204 | 3 = Informational Messages
205 | 4 = All Messages
206 |
207 | You can also pass an array with threshold levels to show individual error types
208 |
209 | array(2) = Debug Messages, without Error Messages
210 |
211 | For a live site you'll usually only enable Errors (1) to be logged otherwise
212 | your log files will fill up very fast.
213 |
214 */
215 $config['log_threshold'] = 0;
216 |
217 /*
218 |-----
219 | Error Logging Directory Path
220 |-----
221 |
222 | Leave this BLANK unless you would like to set something other than the default
223 | application/logs/ directory. Use a full server path with trailing slash.
224 |
225 */
226 $config['log_path'] = '';
227 |
228 /*
229 |-----
230 | Log File Extension
231 |-----
232 |
233 | The default filename extension for log files. The default 'php' allows for
234 | protecting the log files via basic scripting, when they are to be stored
235 | under a publicly accessible directory.
236 |
237 | Note: Leaving it blank will default to 'php'.
238 |
239 */
240 $config['log_file_extension'] = '';
241 |
242 /*
243 |-----
244 | Log File Permissions
245 |-----
246 |
247 | The file system permissions to be applied on newly created log files.
248 |
249 | IMPORTANT: This MUST be an integer (no quotes) and you MUST use octal
250 |           integer notation (i.e. 0700, 0644, etc.)
251 |
252 */
253 $config['log_file_permissions'] = 0644;
254 |
255 /*
256 |-----
257 | Date Format for Logs
258 |-----

```

```

259 | Each item that is logged has an associated date. You can use PHP date
260 | codes to set your own date formatting
261 |
262 |
263 */
264 $config['log_date_format'] = 'Y-m-d H:i:s';
265 /*
266 /**
267 | -----
268 | Error Views Directory Path
269 | -----
270 |
271 | Leave this BLANK unless you would like to set something other than the default
272 | application/views/errors/ directory. Use a full server path with trailing slash.
273 |
274 */
275 $config['error_views_path'] = '';
276 /*
277 /**
278 | -----
279 | Cache Directory Path
280 | -----
281 |
282 | Leave this BLANK unless you would like to set something other than the default
283 | application/cache/ directory. Use a full server path with trailing slash.
284 |
285 */
286 $config['cache_path'] = '';
287 /*
288 | -----
289 | Cache Include Query String
290 | -----
291 |
292 |
293 | Whether to take the URL query string into consideration when generating
294 | output cache files. Valid options are:
295 |
296 | FALSE      = Disabled
297 | TRUE       = Enabled, take all query parameters into account.
298 |             Please be aware that this may result in numerous cache
299 |             files generated for the same page over and over again.
300 |             array('q') = Enabled, but only take into account the specified list
301 |             of query parameters.
302 |
303 */
304 $config['cache_query_string'] = FALSE;
305 /*
306 | -----
307 | Encryption Key
308 | -----
309 |
310 |
311 | If you use the Encryption class, you must set an encryption key.
312 | See the user guide for more info.
313 |
314 | https://codeigniter.com/user\_guide/libraries/encryption.html
315 |
316 */
317 $config['encryption_key'] = '';
318 /*
319 | -----
320 | Session Variables
321 | -----
322 |
323 | 'sess_driver'
324 |
325 |     The storage driver to use: files, database, redis, memcached
326 |
327 | 'sess_cookie_name'
328 |
329 |     The session cookie name, must contain only [0-9a-z_-] characters
330 |
331 | 'sess_expiration'
332 |
333 |     The number of SECONDS you want the session to last.
334 |     Setting to 0 (zero) means expire when the browser is closed.
335 |
336 | 'sess_save_path'
337 |
338 |     The location to save sessions to, driver dependent.
339 |
340 |     For the 'files' driver, it's a path to a writable directory.
341 |     WARNING: Only absolute paths are supported!
342 |
343 |     For the 'database' driver, it's a table name.
344 |     Please read up the manual for the format with other session drivers.
345 |
346 |     IMPORTANT: You are REQUIRED to set a valid save path!
347 |
348 | 'sess_match_ip'
349 |
350 |     Whether to match the user's IP address when reading the session data.
351 |
352 |     WARNING: If you're using the database driver, don't forget to update
353 |             your session table's PRIMARY KEY when changing this setting.
354 |
355 | 'sess_time_to_update'
356 |
357 |

```

```

358 |     How many seconds between CI regenerating the session ID.
359 |
360 | 'sess_regenerate_destroy'
361 |
362 |     Whether to destroy session data associated with the old session ID
363 |     when auto-regenerating the session ID. When set to FALSE, the data
364 |     will be later deleted by the garbage collector.
365 |
366 | Other session cookie settings are shared with the rest of the application,
367 | except for 'cookie_prefix' and 'cookie_httponly', which are ignored here.
368 |
369 */
370 $config['sess_driver'] = 'files';
371 $config['sess_cookie_name'] = 'ci_session';
372 $config['sess_expiration'] = 7200;
373 $config['sess_save_path'] = NULL;
374 $config['sess_match_ip'] = FALSE;
375 $config['sess_time_to_update'] = 300;
376 $config['sess_regenerate_destroy'] = FALSE;
377 /*
378 |-----
379 |-----|Cookie Related Variables|-----|
380 |-----|
382 |
383 | 'cookie_prefix' = Set a cookie name prefix if you need to avoid collisions
384 | 'cookie_domain' = Set to .your-domain.com for site-wide cookies
385 | 'cookie_path' = Typically will be a forward slash
386 | 'cookie_secure' = Cookie will only be set if a secure HTTPS connection exists.
387 | 'cookie_httponly' = Cookie will only be accessible via HTTP(S) (no javascript)
388 |
389 | Note: These settings (with the exception of 'cookie_prefix' and
390 |       'cookie_httponly') will also affect sessions.
391 |
392 */
393 $config['cookie_prefix'] = '';
394 $config['cookie_domain'] = '';
395 $config['cookie_path'] = '/';
396 $config['cookie_secure'] = FALSE;
397 $config['cookie_httponly'] = FALSE;
398 /*
399 |-----
400 |-----|Standardize newlines|-----|
401 |-----|
403 |
404 | Determines whether to standardize newline characters in input data,
405 | meaning to replace \r\n, \r, \n occurrences with the PHP_EOL value.
406 |
407 | This is particularly useful for portability between UNIX-based OSes,
408 | (usually \n) and Windows (\r\n).
409 |
410 */
411 $config['standardize_newlines'] = FALSE;
412 /*
413 |-----
414 |-----|Global XSS Filtering|-----|
415 |-----|
417 |
418 | Determines whether the XSS filter is always active when GET, POST or
419 | COOKIE data is encountered
420 |
421 | WARNING: This feature is DEPRECATED and currently available only
422 |           for backwards compatibility purposes!
423 |
424 */
425 $config['global_xss_filtering'] = FALSE;
426 /*
427 |-----
428 |-----|Cross Site Request Forgery|-----|
429 |-----|
431 |
432 | Enables a CSRF cookie token to be set. When set to TRUE, token will be
433 | checked on a submitted form. If you are accepting user data, it is strongly
434 | recommended CSRF protection be enabled.
435 |
436 | 'csrf_token_name' = The token name
437 | 'csrf_cookie_name' = The cookie name
438 | 'csrf_expire' = The number in seconds the token should expire.
439 | 'csrf_regenerate' = Regenerate token on every submission
440 | 'csrf_exclude_uris' = Array of URIs which ignore CSRF checks
441 */
442 $config['csrf_protection'] = FALSE;
443 $config['csrf_token_name'] = 'csrf_test_name';
444 $config['csrf_cookie_name'] = 'csrf_cookie_name';
445 $config['csrf_expire'] = 7200;
446 $config['csrf_regenerate'] = TRUE;
447 $config['csrf_exclude_uris'] = array();
448 /*
449 |-----
450 |-----|Output Compression|-----|
451 |-----|
452 |
453 | Enables Gzip output compression for faster page loads. When enabled,
454 | the output class will test whether your server supports Gzip.
455 | Even if it does, however, not all browsers support compression
456 | so enable only if you are reasonably sure your visitors can handle it.

```

```

457 | Only used if zlib.output_compression is turned off in your php.ini.
458 | Please do not use it together with httpd-level output compression.
459 |
460 |
461 | VERY IMPORTANT: If you are getting a blank page when compression is enabled it
462 | means you are prematurely outputting something to your browser. It could
463 | even be a line of whitespace at the end of one of your scripts. For
464 | compression to work, nothing can be sent before the output buffer is called
465 | by the output class. Do not 'echo' any values with compression enabled.
466 |
467 */
468 $config['compress_output'] = FALSE;
469 /*
470 |-----
471 |----- Master Time Reference -----
472 |----- 
473 |----- 
474 |
475 | Options are 'local' or any PHP supported timezone. This preference tells
476 | the system whether to use your server's local time as the master 'now'
477 | reference, or convert it to the configured one timezone. See the 'date
478 | helper' page of the user guide for information regarding date handling.
479 |
480 */
481 $config['time_reference'] = 'local';
482 /*
483 |-----
484 |----- Rewrite PHP Short Tags -----
485 |----- 
486 |----- 
487 |
488 | If your PHP installation does not have short tag support enabled CI
489 | can rewrite the tags on-the-fly, enabling you to utilize that syntax
490 | in your view files. Options are TRUE or FALSE (boolean)
491 |
492 | Note: You need to have eval() enabled for this to work.
493 |
494 */
495 $config['rewrite_short_tags'] = FALSE;
496 /*
497 |-----
498 |----- Reverse Proxy IPs -----
499 |----- 
500 |----- 
501 |
502 | If your server is behind a reverse proxy, you must whitelist the proxy
503 | IP addresses from which CodeIgniter should trust headers such as
504 | HTTP_X_FORWARDED_FOR and HTTP_CLIENT_IP in order to properly identify
505 | the visitor's IP address.
506 |
507 | You can use both an array or a comma-separated list of proxy addresses,
508 | as well as specifying whole subnets. Here are a few examples:
509 |
510 | Comma-separated: '10.0.1.200,192.168.5.0/24'
511 | Array: array('10.0.1.200', '192.168.5.0/24')
512 */
513 $config['proxy_ips'] = '';

```

Listing C.3: database.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 /*
4 |-----
5 |----- DATABASE CONNECTIVITY SETTINGS
6 |----- 
7 |
8 | This file will contain the settings needed to access your database.
9 |
10 | For complete instructions please consult the 'Database Connection'
11 | page of the User Guide.
12 |
13 |-----
14 |----- EXPLANATION OF VARIABLES
15 |----- 
16 |
17 | ['dsn'] The full DSN string describe a connection to the database.
18 | ['hostname'] The hostname of your database server.
19 | ['username'] The username used to connect to the database
20 | ['password'] The password used to connect to the database
21 | ['database'] The name of the database you want to connect to
22 | ['dbdriver'] The database driver. e.g.: mysqli.
23 |     Currently supported:
24 |         cubrid, ibase, mssql, mysql, mysqli, oci8,
25 |         odbc, pdo, postgres, sqlite, sqlite3, sqlsrv
26 | ['dbprefix'] You can add an optional prefix, which will be added
27 |             to the table name when using the Query Builder class
28 | ['pconnect'] TRUE/FALSE - Whether to use a persistent connection
29 | ['db_debug'] TRUE/FALSE - Whether database errors should be displayed.
30 | ['cache_on'] TRUE/FALSE - Enables/disables query caching
31 | ['cachedir'] The path to the folder where cache files should be stored
32 | ['char_set'] The character set used in communicating with the database
33 | ['dbcollat'] The character collation used in communicating with the database
34 |             NOTE: For MySQL and MySQLi databases, this setting is only used
35 |                   as a backup if your server is running PHP < 5.2.3 or MySQL < 5.0.7
36 |                   (and in table creation queries made with DB Forge).
37 |             There is an incompatibility in PHP with mysql_real_escape_string() which
38 |                   can make your site vulnerable to SQL injection if you are using a

```

```

39 |         multi-byte character set and are running versions lower than these.
40 |         Sites using Latin-1 or UTF-8 database character set and collation are unaffected.
41 |         ['swap_pre'] A default table prefix that should be swapped with the dbprefix
42 |         ['encrypt'] Whether or not to use an encrypted connection.
43 |
44 |         'mysql' (deprecated), 'sqlsrv' and 'pdo/sqlsrv' drivers accept TRUE/FALSE
45 |         'mysqli' and 'pdo/mysql' drivers accept an array with the following options:
46 |
47 |             'ssl_key'    - Path to the private key file
48 |             'ssl_cert'   - Path to the public key certificate file
49 |             'ssl_ca'     - Path to the certificate authority file
50 |             'ssl_capath' - Path to a directory containing trusted CA certificates in PEM format
51 |             'ssl_cipher'  - List of *allowed* ciphers to be used for the encryption, separated by colons (':')
52 |             'ssl_verify'  - TRUE/FALSE; Whether verify the server certificate or not ('mysqli' only)
53 |
54 |             ['compress'] Whether or not to use client compression (MySQL only)
55 |             ['stricton'] TRUE/FALSE - forces 'Strict Mode' connections
56 |                         - good for ensuring strict SQL while developing
57 |             ['ssl_options'] Used to set various SSL options that can be used when making SSL connections.
58 |             ['failover'] array - A array with 0 or more data for connections if the main should fail.
59 |             ['save_queries'] TRUE/FALSE - Whether to "save" all executed queries.
60 |                         NOTE: Disabling this will also effectively disable both
61 |                         $this->db->last_query() and profiling of DB queries.
62 |                         When you run a query, with this setting set to TRUE (default),
63 |                         CodeIgniter will store the SQL statement for debugging purposes.
64 |                         However, this may cause high memory usage, especially if you run
65 |                         a lot of SQL queries ... disable this to avoid that problem.
66 |
67 |             The $active_group variable lets you choose which connection group to
68 |             make active. By default there is only one group (the 'default' group).
69 |
70 |             The $query_builder variables lets you determine whether or not to load
71 |             the query builder class.
72 */
73 $active_group = 'default';
74 $query_builder = TRUE;
75
76 $db['default'] = array(
77     'dsn'      => '',
78     'hostname'  => 'localhost',
79     'username'  => 'root',
80     'password'  => '',
81     'database'  => 'piktora',
82     'dbdriver'  => 'mysqli',
83     'dbprefix'  => '',
84     'pconnect'  => FALSE,
85     'db_debug'  => (ENVIRONMENT !== 'production'),
86     'cache_on'  => FALSE,
87     'cachedir'  => '',
88     'char_set'  => 'utf8',
89     'dbcollat'  => 'utf8_general_ci',
90     'swap.pre'  => '',
91     'encrypt'   => FALSE,
92     'compress'  => FALSE,
93     'stricton'  => FALSE,
94     'failover'  => array(),
95     'save_queries' => TRUE
96 );

```

Listing C.4: About.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class About extends CI_Controller {
5
6     public function index()
7     {
8         $this->load->view('about');
9     }
10 }

```

Listing C.5: Contact.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Contact extends CI_Controller {
5
6     public function index()
7     {
8         $this->load->helper('string');
9         $this->load->config('piktora');
10        if($this->input->method()=='post'){
11            $expiration = time() - $this->config->item('captcha-settings')['expiration'];
12            $this->db->where('captcha_time < ', $expiration)->delete('captcha');
13            $this->db->where('word', strtolower($this->input->post('captcha')));
14            $this->db->where('ip_address', $this->input->ip_address());
15            $this->db->where('captcha_time >', $expiration);
16            $query = $this->db->get('captcha');
17            if ($query->num_rows() == 0)
18            {
19                $this->session->set_flashdata('warning', "Sorry. Wrong Captcha.");
20            }
21        }else{
22            $this->load->model("Email_model");
23            $name = $this->input->post('name');

```

```

24     $email = $this->input->post('email');
25     $subject = $this->input->post('subject');
26     $tempMessage = $this->input->post('message');
27     $message = "<p>Name: $name</p><p>Email: $email</p><p>Message: $tempMessage</p><p>*note: jika ingin membalas email,
28         balas dengan email tujuan yang tertera di pesan (tidak melalui reply)</p>";
29     $this->Email_model->send_email($email, $name, $subject, $message);
30     $this->session->set_flashdata('success', "Thanks for your response.");
31   }
32   redirect('/contact', 'refresh');
33 }
34 else{
35   $this->load->helper('captcha');
36   $cap = create_captcha($this->config->item('captcha-settings'));
37   $data = array(
38     'captcha_time' => $cap['time'],
39     'ip_address' => $this->input->ip_address(),
40     'word' => strtolower($cap['word'])
41   );
42   $query = $this->db->insert_string('captcha', $data);
43   $this->db->query($query);
44   $this->load->view('contact', array('captcha'=>$cap['image']));
45 }
46 }
47 }

```

Listing C.6: About.php

```

1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class Migrate extends CI_Controller {
6
7 /**
8 * Jika dibuka di http://localhost/migrate akan mengupdate ke db terbaru.
9 * Jika ada data sendiri, mohon dibackup dulu.
10 */
11 public function index() {
12   $this->load->config('migration');
13   if ($this->config->item('migration_enabled') === TRUE) {
14     $this->load->library('migration');
15     set_time_limit(300);
16     $version = $this->migration->latest();
17     if ($version === FALSE) {
18       show_error($this->migration->error_string());
19     } else {
20       echo "Migrate success! DB Version: $version";
21     }
22   } else {
23     echo 'Migration not enabled in config!';
24   }
25 }
26 }
27 }

```

Listing C.7: Projects.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Projects extends CI_Controller {
5
6   public function index()
7   {
8     $query = $this->db->get_where('projects', array(
9       'title IS NOT NULL' => null
10      ));
11     $this->load->view('projects', array('projects' => $query->result_array()));
12   }
13
14   public function detail($id)
15   {
16     $query = $this->db->get_where('projects', array(
17       'id' => $id
18      ));
19     $project = $query->row_array();
20     $photos = [];
21     for ($i = 1; file_exists("assets/img/projects/$id-$i.jpg"); $i++) {
22       $photos[] = "/assets/img/projects/$id-$i.jpg";
23     }
24     $project['photos'] = $photos;
25     $this->load->view('project_details', $project);
26   }
27 }

```

Listing C.8: Welcome.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Welcome extends CI_Controller {
5
6   public function index()
7   {

```

```

8     $photos = [];
9     for ($i = 1; file_exists("assets/img/home/$i.jpg"); $i++) {
10        $photos[] = "/assets/img/home/$i.jpg";
11    }
12    $this->load->view('welcome', array('photos'=>$photos));
13 }
14 }
```

Listing C.9: Auth\_model.php

```

1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class Auth_model extends CI_Model {
6
7     private $client;
8
9     public function __construct() {
10         parent::__construct();
11
12         $this->load->config('auth');
13         $this->client = new Google_Client();
14         $this->client->setHttpClient(new GuzzleHttpClient(array(
15             'base_uri' => Google_Client::API_BASE_PATH, // if doesn't work, try to uncomment this.
16             'verify' => getcwd() . '/..../certs/cacert.pem'
17         )));
18         $this->client->setClientId($this->config->item('google-clientid'));
19         $this->client->setClientSecret($this->config->item('google-clientsecret'));
20         $this->client->setRedirectUri($this->config->item('google-redirecturi'));
21         $this->client->addScope('https://www.googleapis.com/auth/userinfo.email');
22         $this->client->addScope('https://www.googleapis.com/auth/userinfo.profile');
23     }
24
25     /**
26      * Panggil method ini untuk mendapatkan hyperlink untuk melakukan OAuth.
27      * Biasanya digunakan saat akan menampilkan halaman "Login with Google".
28      * @return string URL untuk login
29      */
30     public function createAuthURL() {
31         return $this->client->createAuthUrl();
32     }
33
34     /**
35      * Panggil ini untuk menerima kode autentikasi hasil redirect dari Google,
36      * dan menentukan email dan role user yang berhasil login.
37      * @param string $oauthCode kode oauth, didapat dari parameter GET "code".
38      * @return boolean TRUE selalu.
39      * @throws Exception jika autentikasi gagal (ditolak). Exception message
40      * berisi penjelasan kenapa.
41      */
42     public function authenticateOAuthCode($oauthCode) {
43         $this->client->authenticate($oauthCode);
44         $oauth2Service = new Google_Service_Oauth2($this->client);
45         $userInfo = $oauth2Service->userinfo->get();
46         $email = $userInfo['email'];
47         $name = $userInfo['name'];
48
49         $this->load->model('Configuration_model');
50         $allowedUsers = $this->Configuration_model->getCSV('administrator_emails');
51         if (in_array($email, $allowedUsers)) {
52             $this->session->set_userdata('auth', array(
53                 'email' => $email,
54                 'name' => $name,
55             ));
56         } else {
57             throw new Exception("$email ($name) does not have access to admin page!");
58         }
59     }
60
61     public function getUserInfo() {
62         return $this->session->userdata('auth');
63     }
64
65     public function logout() {
66         $this->session->unset_userdata('auth');
67     }
68 }
69 }
```

Listing C.10: Configuration\_model.php

```

1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class Configuration_model extends CI_Model {
6
7     public function set($id, $value) {
8         $this->db->set('value', $value);
9         $this->db->where('id', $id);
10        $this->db->update('configuration');
11    }
12
13    public function get($id) {
14        $query = $this->db->get_where('configuration', array('id' => $id));
15        $data = $query->row();
```

```

16     if ($data !== NULL) {
17         return $data->value;
18     }
19     return NULL;
20 }
21
22 public function getCSV($id) {
23     return preg_split('/ *, */', $this->get($id));
24 }
25 }

```

Listing C.11: Email\_model.php

```

1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class Email_model extends CI_Model {
6
7     public function send_email($email, $name, $subject, $message, $debug = FALSE) {
8         if ($debug === TRUE) {
9             echo $message;
10            exit();
11        }
12        $this->load->config('secrets');
13        $config = $this->config->item('email-config');
14        $this->load->library('email', $config);
15        $this->email->set_newline("\r\n");
16        $this->email->set_crlf("\r\n");
17        $this->email->from("piktora@mailgun.dnartworks.com.au", "PIKTORA");
18        $this->email->to("hello@piktora.com");
19        $this->email->subject($subject);
20        $this->email->message($message);
21        if (!$this->email->send()) {
22            throw new Exception("Sorry. There was system error when trying to sent your email. Please try again later.");
23        }
24    }
25 }

```

Listing C.12: about.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?><!DOCTYPE html>
4 <html lang="en">
5 <head>
6     <?php $this->load->view('templates/head', array('title' => 'About', 'extra_css' => array('slick', 'slick-theme'))); ?>
7     <style>
8         .one-page-background{
9             background-color:white;
10            background-size:contain;
11            width:100%;
12        }
13
14         .content{
15             padding-left:15px;
16             padding-right:15px;
17         }
18
19         .slick-track > div{
20             outline:none;
21         }
22
23         h4{
24             text-transform: uppercase;
25             line-height: 35px;
26             font-weight: bold;
27             font-family: "Montserrat-Bold";
28         }
29
30         .content-next{
31             text-align:right;
32         }
33
34         .title{
35             text-transform: uppercase;
36             line-height: 25px;
37             font-size: 11px;
38             letter-spacing: 2px;
39             font-weight: bold;
40         }
41
42         .description{
43             font-size:10px;
44             line-height: 20px;
45             letter-spacing: 1px;
46         }
47
48         .content-navigation{
49             letter-spacing: 3px;
50             font-size:12px;
51             cursor: pointer;
52             margin-top: 50px;
53         }
54
55         .content-2 h5{

```

```

56     background-image: url("/assets/img/rectangle-services.png");
57     height:28px;
58     background-size: auto;
59     background-repeat: no-repeat;
60     padding-left:20px;
61     padding-top:7px;
62     font-size:12px;
63     letter-spacing: 3px;
64   }
65
66 .content-2 .row > div{
67   padding-top: 0px;
68   padding-left: 0px;
69   padding-right: 0px;
70 }
71
72 .content-2 .description{
73   max-width: 235px;
74 }
75
76 .slick-slide img{
77   display: inline;
78 }
79
80 .special-description{
81   letter-spacing: 3px;
82   font-weight: bold;
83 }
84
85 @media screen and (min-width: 768px) {
86   h4{
87     line-height: 50px;
88     font-size:28px;
89   }
90   .one-page-background{
91     background-image: url("/assets/img/about/img-background-about.jpg");
92     height:100vh;
93     background-size:contain;
94     background-repeat:no-repeat;
95     background-position:center bottom;
96   }
97
98   .content{
99     padding-left:20px;
100    padding-right:20px;
101  }
102
103  .content-1 div{
104    max-width: 50%;
105    margin-left:50%;
106    text-align: right;
107  }
108  .content-1 div:first-child{
109    margin-left:0%;
110    text-align:left;
111  }
112}
113
114 @media screen and (min-width: 992px) {
115   h4{
116     line-height: 50px;
117     font-size:22px;
118     margin-top: 0px;
119     margin-bottom: 0px;
120   }
121
122   .one-page-background{
123     background-size:cover;
124   }
125   .content{
126     padding-left:60px;
127     padding-right:60px;
128   }
129
130   .content-1 div{
131     width: 25%;
132   }
133
134   .content-1 div:first-child{
135     float:left;
136   }
137
138   .content-1 .content-next{
139     margin-left:75%;
140   }
141
142   .content-1 .special-description{
143     padding-top:18px;
144     width:30%;
145     margin-left:45%;
146   }
147 }
148 </style>
149 </head>
150 <body>
151   <div class="one-page-background">
152     <?php $this->load->view('templates/navbar'); ?>
153     <div class="content">
154       <div class="content-1">

```

```

155     <div>
156         <p class="title">We Produce And Deliver The Highest Possible User Experience Through Innovative Design</p>
157         <p class="description">PIKTORA specializes in creative and marketing material designs, often needed by
158             companies ranging from start-up level to the prominent areas. With solid team of graphic designers,
159             photographers, and web developers, we guarantee a creativity, originality, and identity to your brand.</p>
160     </div>
161     <div class="special-description">
162         <h4>We Do<br>Something Fun<br>To Make<br>Something Special</h4>
163     </div>
164     <div class="content-navigation content-next">
165         <p>WHAT WE DO </p>
166     </div>
167     <div class="content-2">
168         <div class="container-fluid">
169             <div class="row">
170                 <div class="col-xs-12 col-sm-6 col-md-3">
171                     <h5>BRANDING</h5>
172                     <p class="description">It all started with a discovery session where we will chat about your vision
173                         and expectation for your creative needs. Just mention and describe what you have in mind, or we
174                         can even brainstorm and gather ideas together; then we'll explore them from scratch. From
175                         there, leave the rest to us, we'll translate your creative ideas into a workable concept with a
176                         strong impression.</p>
177                 </div>
178                 <div class="col-xs-12 col-sm-6 col-md-3">
179                     <h5>GRAPHIC DESIGN</h5>
180                     <p class="description">Let us know what your target market is. We can design every single marketing
181                         material that you need. We'll act as your artwork consultant to narrow down your needs in order
182                         to achieve your goals. We always take a closer look in every detail, to make sure that the
183                         appearance of your brand looks consistent and meaningful.</p>
184                 </div>
185                 <div class="col-xs-12 col-sm-6 col-md-3">
186                     <h5>WEBSITE DESIGN</h5>
187                     <p class="description">We believe that a good design comes with a good user experience on how people
188                         access a brand in the virtual world. Piktora specializes in developing a brand, while also
189                         collaborating with the current trend. Remember, a website is not only a tool that tells your
190                         audience about what your brand does, but also creates an impressive display in today's modern
191                         industry, as well as an affective user experience from the very beginning.</p>
192                 </div>
193             </div>
194         <div class="content-navigation content-prev">
195             <p> BACK TO OUR STORY</p>
196         </div>
197     </div>
198     </div>
199     <?php $this->load->view('templates/scripts', array('extra_js' => array('slick.min'))); ?>
200 <script>
201     $(document).ready(function(){
202         $(".content").slick({
203             swipe : false,
204             arrows: false
205         });
206         $(".content-next").click(function(event) {
207             $(".content").slick("slickNext");
208         });
209         $(".content-prev").click(function(event) {
210             $(".content").slick("slickPrev");
211         });
212     });
213 </script>
214 </body>
215 </html>

```

Listing C.13: contact.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?<!DOCTYPE html>
4 <html lang="en">
5 <head>
6     <?php $this->load->view('templates/head', array('title' => 'Contact', 'extra_css'=> array('font-awesome.min'))); ?>
7     <style>
8         body{
9             background-color:white;
10        }
11
12         .form-control{
13             color:#474342;
14             background-color:#eeeeae9;
15             border:0px;
16        }
17
18         .captcha-container > div {
19             padding-left:0px;
20             padding-right:0px;
21        }
22
23         .captcha-container div .form-control{
24             height:50px;
25             text-align:center;
26        }
27
28         .captcha-container div:first-child .form-control{
29             background-color:white;

```

```

30         border:1px solid;
31     }
32
33     .send-button{
34         background:none;
35         border:none;
36         box-shadow: none;
37         padding-left:0px;
38         text-transform: uppercase;
39         font-size:11px;
40         letter-spacing: 3px;
41     }
42
43     .send-button:hover{
44         background:none;
45     }
46
47     .captcha-container div:last-child{
48         margin-top:20px;
49     }
50
51     .container-title{
52         padding-right:20%;
53     }
54
55     .title{
56         margin-top:0px;
57         letter-spacing: 2px;
58         font-family: "Montserrat-Bold";
59     }
60
61     .description{
62         font-size:11px;
63         text-transform: uppercase;
64         letter-spacing: 2px;
65         line-height: 25px;
66     }
67
68     #captcha{
69         width:100%;
70         height:50px;
71     }
72
73     @media screen and (min-width: 768px) {
74         .container-contact{
75             padding-left:20px;
76             padding-right:20px;
77         }
78
79         .container-footer{
80             position:absolute;
81             bottom:0;
82         }
83
84         div .captcha-code{
85             padding-right: 10px;
86         }
87
88         div .captcha-input{
89             padding-left: 10px;
90         }
91     }
92
93     @media screen and (min-width: 992px) {
94         .container-contact{
95             padding-left:60px;
96             padding-right:60px;
97         }
98
99         .container-title{
100             padding-right:0px;
101         }
102
103         .captcha-container div:last-child{
104             margin-top:0px;
105         }
106     }
107
108     </style>
109 </head>
110 <body>
111     <?php $this->load->view('templates/navbar'); ?>
112     <?php $this->load->view('templates/flashmessage'); ?>
113     <div class="container-fluid container-contact">
114         <div class="row">
115             <div class="col-xs-12 col-sm-6 col-md-3 container-title">
116                 <h2 class="title">SAY HELLO ,</h2>
117                 <p class="description">DonâŽt hesitate to know us more.</p>
118             </div>
119             <div class="col-xs-12 col-sm-6 col-md-6">
120                 <?= form_open('/contact'); ?>
121                     <div class="form-group">
122                         <input type="text" name="name" class="form-control" placeholder="Your Name" required="required">
123                     </div>
124                     <div class="form-group">
125                         <input type="email" name="email" class="form-control" placeholder="Email Address" required="required">
126                     </div>
127                     <div class="form-group">
128                         <input type="text" name="subject" class="form-control" placeholder="Subject:" required="required">

```

```

129         </div>
130         <div class="form-group">
131             <textarea class="form-control" name="message" rows="7" placeholder="What do you want to say?" required="" required></textarea>
132         </div>
133         <div class="form-group captcha-container">
134             <div class="col-xs-6 col-md-3 captcha-code">
135                 <?=$captcha; ?>
136             </div>
137             <div class="col-xs-6 col-md-3 captcha-input">
138                 <input type="text" name="captcha" class="form-control" placeholder="Enter the code" required="required" required>
139             </div>
140             <div class="col-xs-6 col-md-3 col-md-offset-3">
141                 <button type="submit" class="btn btn-default send-button">Send Now </button>
142             </div>
143             </div>
144             <?= form_close(); ?>
145         </div>
146     </div>
147 </div>
148 </div>
149 <?php $this->load->view('templates/footer'); ?>
150 <?php $this->load->view('templates/scripts'); ?>
151 </body>
152 </html>

```

Listing C.14: projects.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?><!DOCTYPE html>
4 <html lang="en">
5 <head>
6     <?php $this->load->view('templates/head', array('title' => 'Projects'));>
7     <style>
8         .content{
9             color:black;
10            margin-bottom:30px;
11        }
12
13        .content h5{
14            line-height-bottom: 0px;
15            padding-top:10px;
16            line-height: 3px;
17            letter-spacing: 2px;
18            font-size:11px;
19            font-weight: bold;
20        }
21
22        .content p{
23            font-size:10px;
24            letter-spacing: 1px;
25            margin-bottom: 10px;
26        }
27
28        .content .description{
29            text-transform: uppercase;
30            font-size:9px;
31        }
32
33        @media screen and (min-width: 768px) {
34            .content{
35                min-height:300px;
36            }
37        }
38
39        @media screen and (min-width: 992px) {
40            .content{
41                padding-left:6.67%;
42            }
43        }
44    </style>
45 </head>
46 <body>
47     <?php $this->load->view('templates/navbar'); ?>
48     <div class="container container-content">
49         <div class="row">
50             <?php foreach ($projects as $project): ?>
51                 <a href="/projects/<?= $project['id'] ?>">
52                     <div class="col-xs-12 col-sm-6 col-md-4 content">
53                         ">
54                         <h3><?= $project['title'] ?></h3>
55                         <p><?= $project['products'] ?></p>
56                         <p class="description">- <?= $project['categories'] ?></p>
57                     </div>
58                 </a>
59             <?php endforeach; ?>
60         </div>
61     </div>
62     <?php $this->load->view('templates/footer'); ?>
63     <?php $this->load->view('templates/scripts'); ?>
64 </body>
65 </html>

```

Listing C.15: welcome.php

```

1 | <?php
2 | defined('BASEPATH') OR exit('No direct script access allowed');
3 | ?><!DOCTYPE html>
4 | <html lang="en">
5 | <head>
6 |   <?php $this->load->view('templates/head', array('title' => 'Welcome', 'extra_css' => array('slick', 'slick-theme'))); ?>
7 |   <style>
8 |     .banner{
9 |       position:absolute;
10 |       width:100%;
11 |       height:100vh;
12 |       top:0;
13 |       z-index:-100;
14 |     }
15 |
16 |     .banner div {
17 |       background-size:cover;
18 |       background-position:center;
19 |       width:100%;
20 |       height:100%;
21 |     }
22 |
23 |     .text-banner{
24 |       background:transparent;
25 |       padding-left:15px;
26 |       padding-right:25%;
27 |       padding-top:30px;
28 |       padding-bottom:10px;
29 |       color:white;
30 |       font-family: "Montserrat-Light";
31 |       letter-spacing: 2px;
32 |       text-transform: uppercase;
33 |       line-height: 20px;
34 |       font-size:8px;
35 |       color:#fffffa;
36 |     }
37 |
38 |     .container-footer{
39 |       position:absolute;
40 |       bottom:0;
41 |     }
42 |
43 |     @media screen and (min-width: 768px) {
44 |       .text-banner{
45 |         font-size:11px;
46 |         padding-right:15px;
47 |         width:50%;
48 |         background-color:transparent;
49 |         padding-left:20px;
50 |         line-height: 25px;
51 |       }
52 |     }
53 |
54 |     @media screen and (min-width: 992px) {
55 |       .text-banner{
56 |         width:30%;
57 |         padding-left:60px;
58 |       }
59 |     }
60 |   </style>
61 | </head>
62 | <body>
63 |   <?php $this->load->view('templates/navbar', array('menuColor' => 'white')); ?>
64 |   <div class="banner">
65 |     <?php foreach ($photos as $photo): ?>
66 |       <div style="background-image: url(<?= $photo?>);"></div>
67 |     <?php endforeach ?>
68 |   </div>
69 |   <div class="text-banner">
70 |     <p>We are a group of creative people who always have our treats before doing our works. Coffee and sweets are our best friend.</p>
71 |   </div>
72 |   <?php $this->load->view('templates/footer'); ?>
73 |   <?php $this->load->view('templates/scripts', array('extra_js' => array('slick.min'))); ?>
74 |   <script>
75 |     $(document).ready(function(){
76 |       $(".banner").slick({
77 |         autoplay: true,
78 |         autoplaySpeed: 2000,
79 |         arrows: false,
80 |         swipe:false
81 |       });
82 |     });
83 |   </script>
84 | </body>
85 | </html>

```

Listing C.16: project\_details.php

```

1 | <?php
2 | defined('BASEPATH') OR exit('No direct script access allowed');
3 | ?><!DOCTYPE html>
4 | <html lang="en">
5 | <head>
6 |   <?php $this->load->view('templates/head', array('title' => 'Project Details')); ?>
7 |   <style>
8 |     .content-text{

```

```
9|         background-color:#a59c93;
10|         color:#4f4745;
11|         padding-top: 15px;
12|         padding-left: 15px;
13|         padding-bottom: 40px;
14|     }
15|
16|     .content-text a{
17|         color:#4f4745;
18|         text-decoration: underline;
19|         font-weight: bold;
20|     }
21|
22|     .container-content-text{
23|         padding-left: 0px;
24|     }
25|
26|     .container-content-img img{
27|         margin-bottom: 20px;
28|         max-width: 80%;
29|     }
30|
31|     .back-to-previous{
32|         margin-top:20px;
33|     }
34|
35|     .back-to-previous a{
36|         color:#4f4745;
37|         font-weight: bold;
38|         text-transform: uppercase;
39|         letter-spacing: 3px;
40|         font-size:10px;
41|     }
42|
43|     .content-text p{
44|         font-size: 10px;
45|         letter-spacing: 1px;
46|     }
47|
48|     h5{
49|         letter-spacing: 2px;
50|         font-size: 11px;
51|     }
52|
53|     .content-text > div{
54|         margin-bottom: 30px;
55|     }
56|
57|     @media screen and (min-width: 768px) {
58|         .content-text{
59|             padding-top: 30px;
60|             padding-left: 20px;
61|             padding-bottom: 250px;
62|         }
63|
64|         .container-content-img img{
65|             max-width: 100%;
66|             padding-right:10px;
67|         }
68|
69|         .back-to-previous{
70|             padding-left:50px;
71|         }
72|
73|         .container-content-img{
74|             overflow-y: auto;
75|         }
76|
77|         .container-content-text{
78|             overflow-y: auto;
79|         }
80|     }
81|
82|     @media screen and (min-width: 992px) {
83|         .container-content-text{
84|             padding-right:80px;
85|         }
86|
87|         .content-text{
88|             padding-left: 60px;
89|             padding-right:60px;
90|             padding-bottom: 100px;
91|         }
92|
93|         .container-content-img img{
94|             padding-right:50px;
95|         }
96|
97|     }
98| 
```

```

108         </div>
109     <div class="col-xs-12 col-sm-6 col-sm-pull-6 col-md-5 col-md-pull-7 container-content-text">
110         <div class="content-text">
111             <div>
112                 <h5><?= $title ?></h5>
113                 <p><?= $products ?></p>
114                 <p>- <?= $categories ?></p>
115             </div>
116             <div>
117                 <h5>About Client</h5>
118                 <p><?= $aboutClient ?></p>
119             </div>
120             <div>
121                 <h5>Our Concept</h5>
122                 <p><?= $ourConcept ?></p>
123             </div>
124             <?= $additionalInfo ?>
125         </div>
126         <div class="back-to-previous">
127             <p><a href="/projects"> See All Projects</a></p>
128         </div>
129     </div>
130 </div>
131 <?php $this->load->view('templates/scripts'); ?>
132 <script>
133     $(document).ready(function() {
134         function setContentHeight(){
135             var height = $(window).height() - $(".navigation").height() - 50;
136             $('.container-content-img').css("height", (height + 'px'));
137             $('.container-content-text').css("height", (height + 'px'));
138         }
139         if($window.width()>=768){
140             setContentHeight();
141         }
142         $(window).resize(function() {
143             if($window.width()>=768){
144                 setContentHeight();
145             }
146         });
147     });
148 });
149 </script>
150 </body>
151 </html>
152

```

Listing C.17: flashmessage.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?>
4 <?php if (isset($_SESSION['success'])): ?>
5     <div class="alert alert-success">
6         <i class="fa fa-check-circle-o"></i> <?= strip_tags($_SESSION['success']) ?>
7     </div>
8 <?php endif; ?>
9
10 <?php if (isset($_SESSION['info'])): ?>
11     <div class="alert alert-info">
12         <i class="fa fa-info-circle"></i> <?= strip_tags($_SESSION['info']) ?>
13     </div>
14 <?php endif; ?>
15
16 <?php if (isset($_SESSION['warning'])): ?>
17     <div class="alert alert-warning">
18         <i class="fa fa-exclamation-triangle"></i> <?= strip_tags($_SESSION['warning']) ?>
19     </div>
20 <?php endif; ?>
21
22 <?php if (isset($_SESSION['danger'])): ?>
23     <div class="alert alert-danger">
24         <i class="fa fa-exclamation-circle"></i> <?= strip_tags($_SESSION['danger']) ?>
25     </div>
26 <?php endif; ?>

```

Listing C.18: footer.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?><!--footer-->
4 <footer class="container-fluid container-footer">
5     <div class="row">
6         <div class="col-xs-12 col-sm-3 col-md-2">
7             <h5>Studio</h5>
8             <p>PT DNARTWORKS KOMUNIKASI VISUAL. 3rd Floor,<br>Ruko Graha Boulevard Blok A no 9,<br>Jl. Pondok Hijau Golf Raya,
9 Gading Serpong, Tangerang, Banten 15810.</p>
10        </div>
11        <div class="col-xs-12 col-sm-3 col-md-2">
12            <h5>hello@piktoria.com</h5>
13            <p>+62 21 2941 8990</p>
14        </div>
15        <div class="col-xs-12 col-sm-3 col-sm-offset-3 col-md-2 col-md-offset-6">
16            <table class="daily-stuff-container">
17                <tbody>
18                    <tr>

```

```

19         <td class="social-media"><a target="_blank" href="https://www.instagram.com/piktorastudio/"></a></td>
21     </tr>
22   </tbody>
23   <p>Copyright &copy; 2017&ampnbsp&nbsp;<b class="product-title">PIKTORA</b></p>
24 </div>
25 </div>
26 </footer>
27 <!--/footer-->
```

Listing C.19: head.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?><!--head-->
4 <meta charset="utf-8">
5 <meta http-equiv="X-UA-Compatible" content="IE=edge">
6 <meta name="viewport" content="width=device-width, initial-scale=1">
7 <title>?= $title ?> | Piktora</title>
8 <link href="/assets/css/bootstrap.min.css" rel="stylesheet">
9 <link href="/assets/css/bootstrap-theme.min.css" rel="stylesheet">
10 <?php if (isset($extra_css)): ?>
11     <?php foreach ($extra_css as $css): ?>
12         <link href="/assets/css/<?= $css ?>.css" rel="stylesheet">
13     <?php endforeach; ?>
14 <?php endif; ?>
15 <link href="/assets/css/style.css" rel="stylesheet">
16 <!--[if lt IE 9]>
17     <script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
18     <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
19 <![endif]-->
20 <!--/head-->
```

Listing C.20: navbar.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 if (!isset($menuColor)){
4     $menuColor='dark';
5 }
6 ?><!--navbar-->
7 <table class="navigation">
8     <tbody>
9         <tr>
10            <td class="navigation-left"><a href="/welcome"></
11                a></td>
12            <td class="navigation-right"></td>
13        </tbody>
14    </table>
15    <div class="menu-fullscreen">
16        <div class="menu-navigation-close">
17            <a id="close-menu" href="#"></a>
18        </div>
19        <div class="menu-navigation-content">
20            <p>Hello, we are Piktora.</p>
21            <p>Know more <a href="/about">about us</a>, </p>
22            <p>explore <a href="/projects">our works</a>, and</p>
23            <p>of course <a href="/contact">say hello</a> to</p>
24            <p>us! We'd love to meet up</p>
25            <p>over coffee (or beer)!</p>
26        </div>
27    </div>
28 <!--/navbar-->
```

Listing C.21: scripts.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?><!--scripts-->
4 <script src="/assets/js/jquery.min.js"></script>
5 <script src="/assets/js/bootstrap.min.js"></script>
6 <?php if (isset($extra_js)): ?>
7     <?php foreach ($extra_js as $js): ?>
8         <script src="/assets/js/<?= $js ?>.js"></script>
9     <?php endforeach; ?>
10 <?php endif; ?>
11 <script src="/assets/js/script.js"></script>
12 <!--/scripts-->
```