

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**

LEMBAR PENGESAHAN

PEMBANGKIT TIMELAPSE PENGEMBANGAN PROYEK PERANGKAT LUNAK BERBASIS WEB

Billy Adiwijaya

NPM: 2015730053

Bandung, «**tanggal**» «**bulan**» 2019

Menyetujui,

Pembimbing

Pascal Alfadian, M.Comp.

Ketua Tim Penguji

Anggota Tim Penguji

Kristopher David Harjono, M.T.

Elisati Hulu, M.T.

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

PEMBANGKIT TIMELAPSE PENGEMBANGAN PROYEK PERANGKAT LUNAK BERBASIS WEB

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal «tanggal» «bulan» 2019

Meterai Rp. 6000

Billy Adiwijaya
NPM: 2015730053

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 *commit history*. Dengan adanya *commit*, pengembangan 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 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 System. 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 shows that software can run well on several WebDriver. In EdgeDriver, layout of the page was broken. 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

Teknik Informatika UNPAR dan diri sendiri

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas seluruh berkat yang diberikan kepada penulis sehingga dapat menyelesaikan tugas akhir dengan judul **Pembangkit Timelapse Pengembangan Proyek Perangkat Lunak Berbasis Web** dengan baik dan tepat waktu. Penulis juga berterima kasih kepada pihak-pihak yang telah memberikan dukungan dan bantuan kepada penulis dalam menyelesaikan tugas akhir ini, yaitu:

1. Keluarga yang selalu memberikan dukungan kepada penulis.
2. Bapak Pascal Alfadian sebagai dosen pembimbing yang telah membimbing penulis hingga dapat menyelesaikan skripsi ini.
3. Bapak Kristopher David Harjono dan Bapak Elisati Hulu sebagai dosen penguji yang telah membantu dalam menguji dan memperbaiki skripsi ini.
4. Teman-teman LKM 2017/2018 yang telah memberikan semangat kepada penulis.
5. Keluarga besar Own Games yang telah memberikan semangat kepada penulis.
6. Pihak-pihak lain yang belum disebutkan, yang berperan dalam penyelesaian tugas akhir ini.

Bandung, «bulan» 2019

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
1 PENDAHULUAN	1
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	3
2 LANDASAN TEORI	5
2.1 Version Control Systems	5
2.1.1 Local Version Control Systems	5
2.1.2 Centralized Version Control Systems	6
2.1.3 Distributed Version Control Systems	7
2.2 Git	7
2.2.1 Cara Kerja Git	7
2.2.2 Operasi Dasar pada Git	10
2.2.3 Git Checkout	12
2.3 JGit	12
2.3.1 Kelas Repository	12
2.3.2 Kelas Ref	13
2.3.3 Kelas FileRepository	13
2.3.4 Kelas Git	13
2.3.5 Kelas CheckoutCommand	15
2.3.6 Kelas ListBranchCommand	15
2.3.7 Enum ListBranchCommand.ListMode	15
2.3.8 Kelas LogCommand	15
2.3.9 Kelas ResetCommand	16
2.3.10 Kelas ResetCommand.ResetType	16
2.3.11 Kelas RevCommit	16
2.3.12 Kelas PersonIdent	17
2.4 Selenium WebDriver	17
2.4.1 Interface WebDriver	18
2.4.2 Interface WebDriver.Options	18
2.4.3 Interface WebDriver.Window	19
2.4.4 Kelas ChromeDriver	19

2.4.5	Kelas FirefoxDriver	19
2.4.6	Kelas OperaDriver	19
2.4.7	Kelas SafariDriver	19
2.4.8	Kelas InternetExplorerDriver	19
2.4.9	Kelas EdgeDriver	19
2.4.10	Kelas HtmlUnitDriver	20
2.4.11	Interface OutputType	20
2.4.12	Interface TakesScreenshot	20
2.5	Apache Commons CLI	20
2.5.1	Interface CommandLineParser	20
2.5.2	Kelas DefaultParser	21
2.5.3	Kelas CommandLine	21
2.5.4	Kelas Options	22
2.5.5	Kelas Option	22
2.5.6	Kelas Option.Builder	22
3	ANALISIS	25
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	33
3.4.1	Perbedaan Letak <i>File</i>	36
3.4.2	Permasalahan Konfigurasi <i>Database</i>	36
3.4.3	Permasalahan Migrasi <i>Database</i>	37
4	PERANCANGAN	39
4.1	Perancangan Kelas	39
4.2	Perancangan Antarmuka	42
5	IMPLEMENTASI DAN PENGUJIAN	45
5.1	Implementasi	45
5.1.1	Lingkungan Implementasi	45
5.1.2	Hasil Implementasi	45
5.2	Pengujian	47
5.2.1	Pengujian Fungsional	47
5.2.2	Pengujian Eksperimental	52
6	KESIMPULAN DAN SARAN	63
6.1	Kesimpulan	63
6.2	Saran	63
DAFTAR REFERENSI	65	
A KODE PROGRAM	67	
B LIBRARY UNTUK MEMBUAT FILE GIF	75	
C KODE PROGRAM PROYEK PIKTORA PADA COMMIT 89000BE (12 JANUARI 2018)	79	

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 directory</i> , <i>Staging area</i> , dan Git direktori[1].	9
2.7 <i>Checkout</i> pada <i>commit</i>	12
3.1 Visualisasi dari repositori perangkat lunak menggunakan Gource.	25
3.2 <i>Commit history</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
3.5 Langkah-langkah untuk membangkitkan animasi <i>timelapse</i>	33
4.1 Diagram kelas.	41
4.2 Rancangan tampilan <i>screenshot</i> pada <i>file</i> hasil animasi jika terdapat satu halaman <i>web</i>	43
4.3 Rancangan tampilan <i>screenshot</i> pada <i>file</i> hasil animasi jika terdapat dua halaman <i>web</i>	43
4.4 Rancangan tampilan <i>screenshot</i> pada <i>file</i> hasil animasi jika terdapat tiga halaman <i>web</i>	44
4.5 Rancangan tampilan <i>screenshot</i> pada <i>file</i> hasil animasi jika terdapat empat halaman <i>web</i>	44
5.1 <i>Screenshot</i> halaman <i>web</i> setiap <i>commit</i> pada situs <i>web</i> Piktora.	46
5.2 Salah satu <i>screenshot</i> halaman <i>web</i> yang terdapat pada <i>file</i> hasil animasi. Terdapat judul di bagian pojok kiri bawah.	47
5.3 Salah satu <i>screenshot</i> halaman <i>web</i> yang terdapat pada <i>file</i> hasil animasi. Terdapat logo di bagian pojok kanan bawah.	47
5.4 Salah satu <i>screenshot</i> halaman <i>web</i> pada <i>file</i> hasil animasi jika terdapat satu argumen <i>-capture-url</i>	51
5.5 Salah satu <i>screenshot</i> halaman <i>web</i> hasil animasi jika terdapat dua argumen <i>-capture-url</i>	51
5.6 Salah satu <i>screenshot</i> halaman <i>web</i> hasil animasi jika terdapat tiga argumen <i>-capture-url</i>	51
5.7 Salah satu <i>screenshot</i> halaman <i>web</i> pada <i>file</i> hasil animasi jika terdapat empat argumen <i>-capture-url</i>	52
5.8 Tampilan pada Firefox <i>browser</i> saat dikontrol oleh FirefoxDriver.	53
5.9 Tampilan pada Opera <i>browser</i> saat dikontrol oleh OperaDriver.	54
5.10 Tampilan halaman <i>web</i> pada <i>browser</i> saat dikontrol oleh EdgeDriver.	55
5.11 Tampilan halaman <i>web</i> pada <i>browser</i> saat tidak dikontrol oleh EdgeDriver.	55
5.12 Keadaan awal dari <i>browser</i> saat dikontrol oleh InternetExplorerDriver.	56
5.13 Tampilan pada <i>browser</i> saat dikontrol oleh InternetExplorerDriver.	56
5.14 Sebagian hasil animasi dari situs <i>web</i> Netflix Open Source Software Center.	57
5.15 Sebagian hasil animasi dari situs <i>web</i> Bootstrap.	58

5.16 Sebagian hasil animasi dari situs <i>web</i> IBM Open Source.	59
5.17 Sebagian hasil animasi dari situs <i>web</i> React.	60
5.18 Sebagian hasil animasi dari situs <i>web</i> Yelp.	61

DAFTAR TABEL

3.1	<i>Scenario case</i> membangkitkan animasi <i>timelapse</i>	32
5.1	Tabel pengujian fungsional	48
5.2	Tabel pengujian fungsional	49
5.3	Tabel pengujian fungsional	50
5.4	Tabel hasil pengujian eksperimental pada beberapa situs <i>web</i>	61

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Git merupakan perangkat lunak *Version Control Systems*[1]. *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 *commit history*. Setiap *commit* mengandung informasi mengenai perubahan yang terjadi pada repositori, waktu perubahan, dan orang yang melakukan perubahan. *Database* pada *git* tidak bersifat terpusat, melainkan terdistribusi. Setiap orang yang terlibat mempunyai *database* lokal pada masing-masing komputer, sehingga pengelolaan perangkat lunak dapat dilakukan secara *online* dan *offline*.

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

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

Dengan adanya *commit history* pada Git, pengembangan suatu proyek perangkat lunak dapat dipantau. *Programmer* dapat melihat kembali secara garis besar langkah-langkah yang sudah dilakukan pada *commit* sebelumnya. *Commit history* juga dapat digunakan untuk mengevaluasi pengembangan perangkat lunak. Akan tetapi, untuk perangkat lunak yang memiliki banyak *commit*, pemantauan progres dapat memakan waktu yang lama. Karena itu proses pengembangan perangkat lunak perlu divisualisasikan.

Timelapse berasal dari kata latin *lapsus* yang bermakna "waktu yang berlalu"[4]. Secara umum *timelapse* dalam fotografi menunjukkan waktu yang lebih cepat dibandingkan dengan waktu sebenarnya. *Timelapse* bisa digunakan untuk mengompres waktu. Misalnya pembangunan suatu gedung dalam waktu tiga tahun bisa ditampilkan dalam bentuk *timelapse* yang hanya berdurasi dua menit.

Pada skripsi ini, dibuat sebuah perangkat lunak yang dapat membangun animasi *timelapse* dari pengembangan proyek perangkat lunak berbasis *web*. Yang dibuat animasinya adalah halaman-halaman *web* dari suatu perangkat lunak berbasis *web* yang terekam oleh Git. Perangkat lunak ini dibangun menggunakan bahasa Java. Perangkat lunak ini menggunakan antarmuka terminal/konsol. Masukan perangkat lunak diambil dari argumen *command line*. Perangkat lunak dibuat menggunakan bantuan *library* JGit dan Selenium WebDriver. *Output* yang dihasilkan dari program ini adalah *file* hasil animasi yang bertipe GIF.

1.2 Rumusan Masalah

Rumusan masalah dari skripsi ini adalah sebagai berikut:

1. Bagaimana cara membangkitkan animasi *timelapse* pada pengembangan proyek perangkat lunak berbasis web?
2. Bagaimana membangun aplikasi untuk membangkitkan *timelapse* pada pengembangan proyek perangkat lunak berbasis web?
3. Kesimpulan apa yang dapat diambil dari animasi *timelapse* pada beberapa situs *web* Open Source?

1.3 Tujuan

Tujuan dari skripsi ini adalah sebagai berikut:

1. Mengetahui cara untuk membangkitkan animasi *timelapse* pada pengembangan proyek perangkat lunak berbasis web.
2. Membangun aplikasi untuk membangkitkan *timelapse* pada pengembangan proyek perangkat lunak berbasis web.
3. Mendapatkan kesimpulan dari animasi *timelapse* pada beberapa situs *web* Open Source.

1.4 Batasan Masalah

Batasan masalah pada skripsi ini adalah sebagai berikut:

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

1.5 Metodologi

Metodologi penelitian yang digunakan dalam skripsi ini adalah sebagai berikut:

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

1.6 Sistematika Pembahasan

Setiap bab dalam penelitian ini memiliki sistematika penulisan yang dijelaskan ke dalam poin-poin sebagai berikut:

1. Bab 1: Pendahuluan, yaitu membahas mengenai gambaran umum penelitian ini. Berisi tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metode penelitian, dan sistematika penulisan.
2. Bab 2: Dasar Teori, yaitu membahas mengenai teori-teori yang mendukung berjalannya penelitian ini. Berisi tentang teori Version Control Systems, Git, JGit, Selenium WebDriver, dan Apache Commons CLI.
3. Bab 3: Analisis, yaitu membahas mengenai analisa masalah. Berisi tentang analisis aplikasi sejenis, analisis penggunaan JGit dan Selenium WebDriver untuk membangkitkan animasi *timelapse*, analisis fitur aplikasi yang dibangun, dan prapengujian.
4. Bab 4: Perancangan, yaitu membahas mengenai perancangan yang dilakukan sebelum melakukan tahapan implementasi. Berisi tentang perancangan kelas dan perancangan antarmuka dari perangkat lunak.
5. Bab 5: Implementasi dan Pengujian, yaitu membahas mengenai implementasi dan pengujian aplikasi yang telah dilakukan. Berisi tentang implementasi dan hasil pengujian perangkat lunak.
6. Bab 6: Kesimpulan dan Saran, yaitu membahas hasil kesimpulan dari keseluruhan penelitian ini dan saran-saran yang dapat diberikan untuk penelitian berikutnya.

BAB 2

LANDASAN TEORI

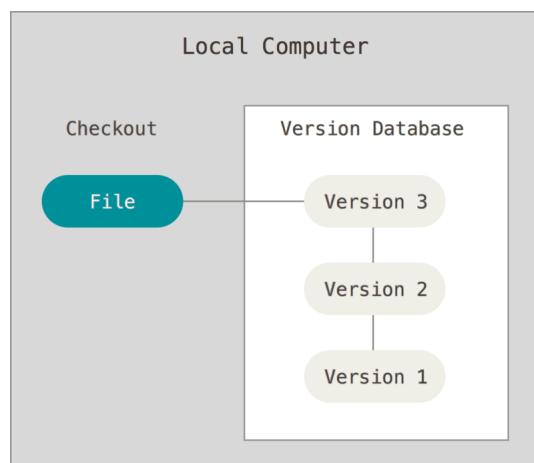
Pada bab ini dibahas dasar teori yang mendukung skripsi ini. Dasar teori yang dibahas yaitu Version Control Systems, Git, JGit, Selenium WebDriver, dan Apache Commons CLI.

2.1 Version Control Systems

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

2.1.1 Local Version Control Systems

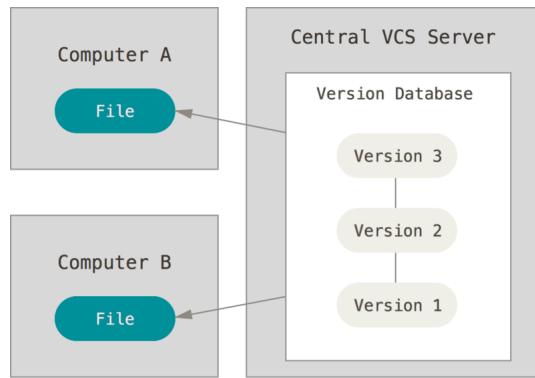
Metode pengendalian versi yang biasa digunakan adalah dengan cara menyalin sekumpulan *file* ke direktori lain. Namun cara tersebut rentan terhadap *error*. Misalnya, terdapat direktori A dan B, pengguna ingin mengubah *file* yang terdapat pada direktori B, tetapi pengguna lupa kalau dia sedang berada di direktori A, maka pengguna mengubah *file* pada direktori yang salah. Untuk mengatasi masalah tersebut, dikembangkanlah *Local Version Control Systems*.



Gambar 2.1: Local version control[1].

Gambar 2.1 merupakan struktur dari *Local Version Control Systems*. *Database local Version Control Systems* ini tersimpan pada direktori lokal di komputer. *Database* ini menyimpan perubahan *file* ke dalam beberapa versi.

2.1.2 Centralized Version Control Systems



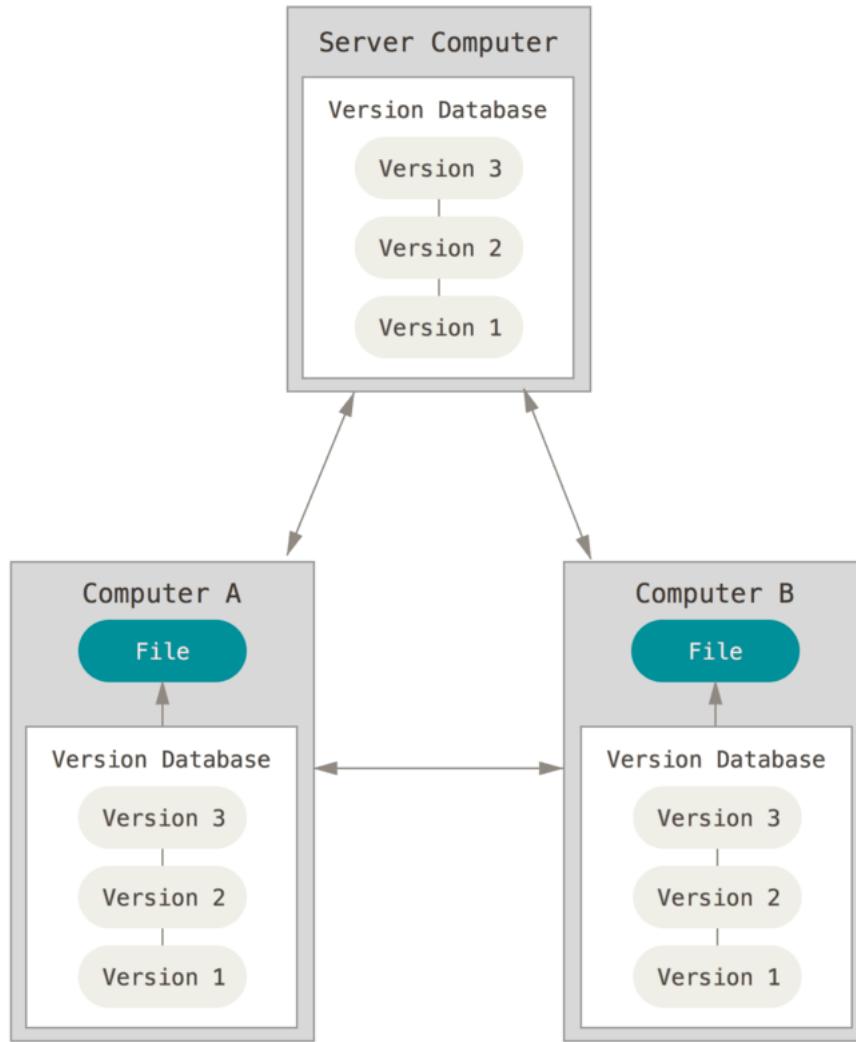
Gambar 2.2: Centralized version control[1].

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

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

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

2.1.3 Distributed Version Control Systems



Gambar 2.3: Distributed version control[1].

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*.

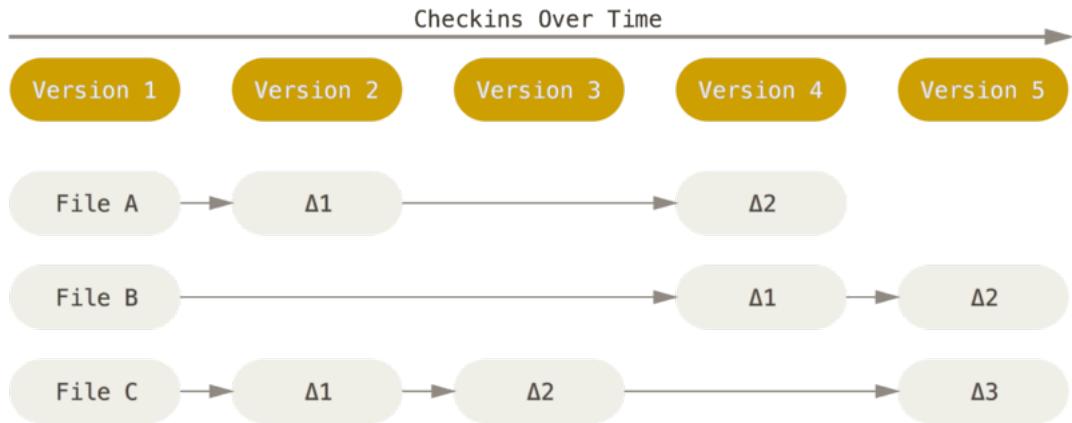
2.2 Git

Pada subbab ini, dijelaskan mengenai cara kerja Git, Git *checkout*, dan operasi-operasi dasar pada Git. Subbab ini mengacu pada [1].

2.2.1 Cara Kerja Git

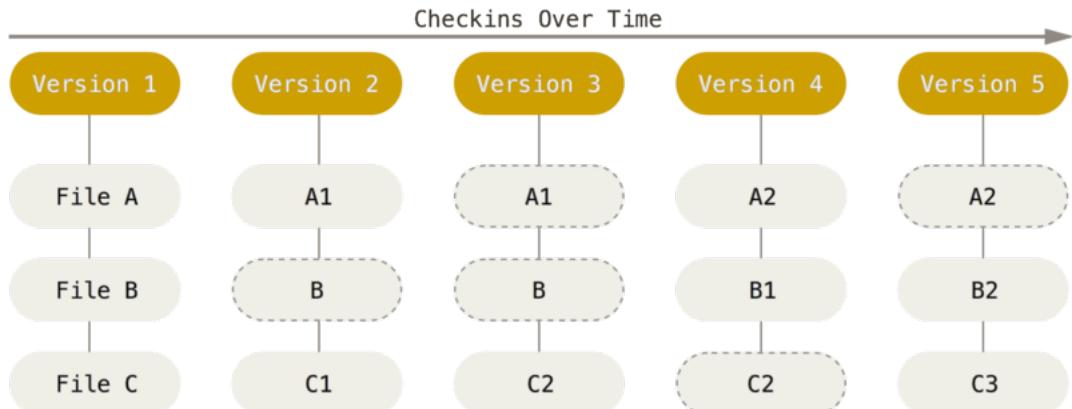
Salah satu perbedaan antara Git dengan VCS lainnya adalah dalam cara Git memperlakukan datanya. Kebanyakan sistem *Version Control Systems* lain menyimpan informasi sebagai daftar

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].

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



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

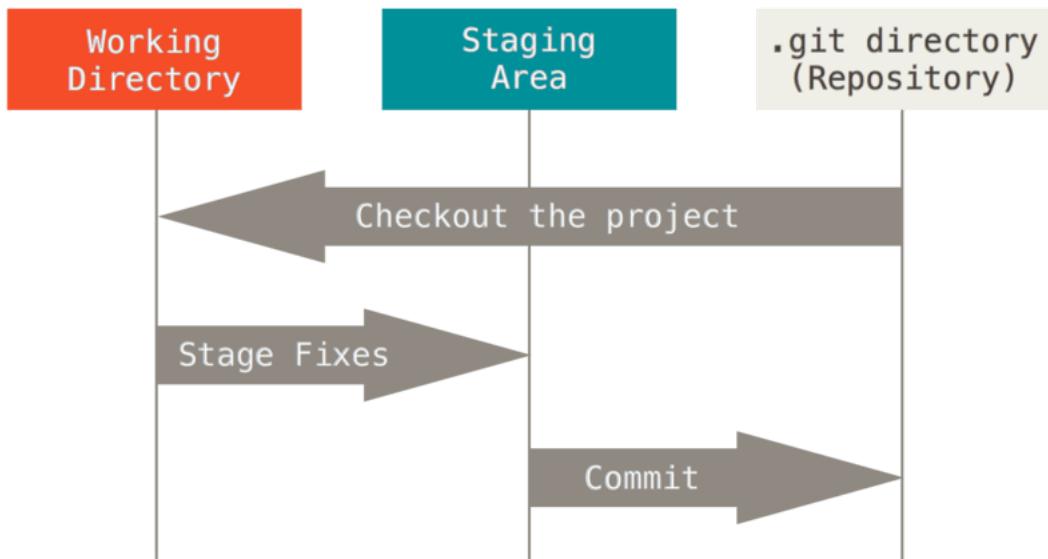
State pada Git

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

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

Alur kerja dari Git adalah sebagai berikut:

1. Melakukan modifikasi pada *file*.
2. Menandai perubahan pada *file* dan memindahkannya ke *staging area*.
3. Mengambil *file* dari *staging area* dan menyimpan *snapshot* ke direktori Git. Proses ini disebut dengan *commit*.



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

Commit

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 *commit history* dalam pengembangan perangkat lunak

```

1 commit 89000be7ce7d16f006813cddefb4ec6d70d15ed6 (HEAD -> master, origin/master,
  origin/HEAD)
2 Author: Hizkia Steven <xvii.hs@gmail.com>
3 Date:   Fri Jan 12 12:25:30 2018 +0700
4
5     Update new company address
6
7 commit 6a085c1c37949e6308cfe06a117302e528388e54
8 Author: Hizkia Steven <xvii.hs@gmail.com>
9 Date:   Tue Dec 12 14:38:38 2017 +0700
10
11    Update company address
12
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 *commit history* 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*.

2.2.2 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 akan 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>]
```

4. Branch

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

```

$ git branch [-a]
$ git branch [-r]
$ git branch [branch-name]
$ git branch [-d <branch-name>]
$ git branch [-D <branch-name>]

```

5. Diff

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

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

6. Clone

Operasi ini digunakan untuk menyalin *remote repository* ke repositori lokal. Berikut adalah sintaks untuk melakukan operasi *clone*:

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

7. Fetch

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

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

8. Merge

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

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

9. Pull

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

```
$ git pull
```

10. Push

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

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

11. Checkout

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

```
$ git checkout [commit ID]  
$ git checkout [branch-name]
```

12. Log

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

```
$ git log
```

13. Reset

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

- Soft

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

- Hard

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

- Mixed

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

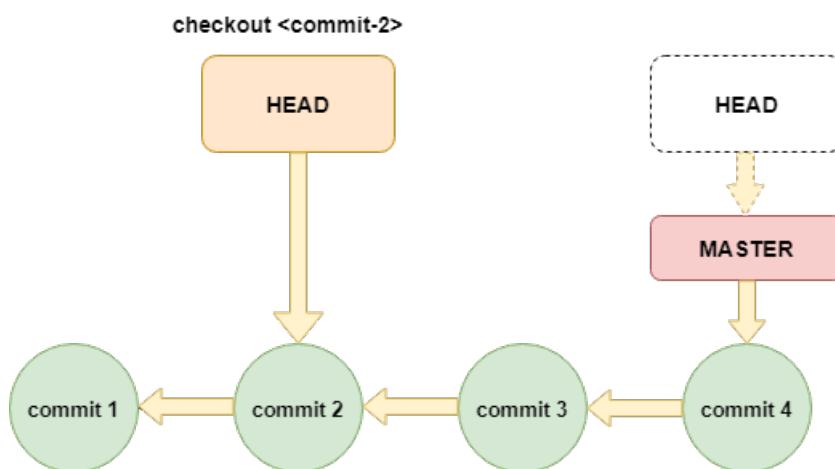
Berikut ini adalah sintaks untuk melakukan operasi *reset*:

```
$ git reset -hard [commit]
$ git reset -mixed [commit]
$ git reset -soft [commit]
```

2.2.3 Git Checkout

Seperti yang sudah dijelaskan pada subbab 2.2.2, *checkout* dapat digunakan untuk berpindah ke *branch* atau *commit* tertentu. Operasi *checkout* dapat dilakukan menggunakan sintaks `$ git checkout` diikuti dengan nama *branch* atau *commit ID*. Gambar 2.7 menunjukkan contoh *checkout* pada *commit*. Posisi awal *HEAD* menunjuk pada *branch master*, setelah dilakukan *checkout* ke *commit 2*, posisi *HEAD* menunjuk pada *commit 2*. *Working directory* diperbarui berdasarkan *state* pada *commit 2*.

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



Gambar 2.7: *Checkout* pada *commit*

2.3 JGit

JGit adalah *library* Java murni yang mengimplementasikan Git *version control systems*[2]. Dengan menggunakan JGit, operasi-operasi dalam Git bisa dilakukan melalui program Java. Pada subbab berikut dijelaskan beberapa kelas dari *library* JGit. Subbab ini mengacu pada [5].

2.3.1 Kelas Repository

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

- public void create()
Berfungsi untuk membuat repositori Git baru.
- public String getBranch()
Berfungsi untuk mendapatkan nama *branch* yang ditunjuk oleh *HEAD*.
Kembalian: nama dari *branch* yang sedang aktif.
- public Ref getRef(String name)
Berfungsi untuk mendapatkan *reference* berdasarkan nama yang diberikan.
Parameter: nama dari *reference*.
Kembalian: *object* bertipe Ref.

2.3.2 Kelas Ref

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

- String getName()
Berfungsi untuk mengembalikan nama dari suatu Ref.
Kembalian: nama dari Ref.
- ObjectId getObjectId()
Berfungsi untuk mengembalikan ID dari suatu Git Object.
Kembalian: *object* bertipe ObjectId. Di dalam kelas ObjectId terdapat *method* getName() yang berfungsi untuk mengembalikan ID dari Git Object berupa SHA-1 *hash value*.

2.3.3 Kelas FileRepository

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

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

2.3.4 Kelas Git

Kelas ini menyediakan API yang mirip Git *command line* untuk berinteraksi dengan repositori Git. Berikut ini adalah *constructor* dan beberapa *method* dalam kelas ini:

- public Git(Repository repo)
Constructor ini membuat objek Git yang digunakan untuk berinteraksi dengan repositori Git.
Parameter: objek *Repository* yang digunakan untuk berinteraksi. Parameter tidak boleh bernilai *null*.
- public static InitCommand init()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *init*.
Kembalian: objek *InitCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *init*.
- public AddCommand add()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *add*.
Kembalian: objek *AddCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *add*.

- public LogCommand log()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *log*.
Kembalian: objek *LogCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *log*.
- public CheckoutCommand checkout()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *checkout*.
Kembalian: objek *ChecoutCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *checkout*.
- public CommitCommand commit()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *commit*.
Kembalian: objek *CommitCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *commit*.
- public FetchCommand fetch()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *fetch*.
Kembalian: objek *FetchCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *fetch*.
- public PushCommand push()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *push*.
Kembalian: objek *PushCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *push*.
- public DiffCommand diff()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *diff*.
Kembalian: objek *DiffCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *diff*.
- public static CloneCommand cloneRepository()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *clone*.
Kembalian: objek *CloneCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *clone*.
- public MergeCommand merge()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *merge*.
Kembalian: objek *MergeCommand* yang berfungsi untuk mengumpulkan parameter opsional dan akhirnya mengeksekusi operasi *merge*.
- public PullCommand pull()
Method ini mengembalikan objek *command* untuk mengeksekusi operasi *pull*.
Kembalian: objek *PullCommand*.
- public CreateBranchCommand branchCreate()
Method ini mengembalikan objek *command* untuk membuat *branch* baru.
Kembalian: objek *CreateBranchCommand*.
- public ListBranchCommand branchList()
Method ini mengembalikan objek *command* yang digunakan untuk mengeksekusi operasi *branch*. Operasi ini setara dengan sintaks *git branch -r/-a*.
Kembalian: objek dengan tipe *ListBranchCommand*.
- public DeleteBranchCommand branchDelete()
Method ini mengembalikan objek *command* untuk menghapus *branch*.
Kembalian: objek *DeleteBranchCommand*.

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

2.3.5 Kelas CheckoutCommand

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

- public CheckoutCommand setName(String name)
Menentukan nama *branch* atau *commit* ID untuk melakukan checkout.
Parameter: nama dari *branch* atau *commit* ID.
Kembalian: *object* *ChecoutCommand*.
- public Ref call()
Berfungsi untuk menjalankan operasi *checkout*.

2.3.6 Kelas ListBranchCommand

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

- public ListBranchCommand setListMode(ListBranchCommand.ListMode listMode)
Menentukan mode yang digunakan untuk mendapatkan *branch*. Jika parameter tidak dimasukkan, hanya daftar *branch* lokal yang didapatkan.
Parameter: mode yang digunakan untuk mendapatkan *branch*, parameter ini berupa konstanta.
Kembalian: *object* *ChecoutCommand*.
- public List<Ref> call()
Berfungsi untuk menjalankan operasi *branch*.
Kembalian: daftar *branch* dalam bentuk List<Ref>.

2.3.7 Enum ListBranchCommand.ListMode

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

- public static final ListBranchCommand.ListMode ALL
Mode ini digunakan untuk mendapatkan semua daftar *branch* baik lokal maupun *remote*.
Mode ini setara dengan *option -a* pada sintaks `git branch -a`.
- public static final ListBranchCommand.ListMode REMOTE
Mode ini digunakan untuk mendapatkan daftar *remote branch*. Mode ini setara dengan *option -r* pada sintaks `git branch -r`.

2.3.8 Kelas LogCommand

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

- public Iterable<RevCommit> call()
Berfungsi untuk menjalankan operasi *log*.
Kembalian: *commit history* pada *branch* yang sedang aktif.

2.3.9 Kelas ResetCommand

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

- public ResetCommand setMode(ResetCommand.ResetType mode)
Menentukan tipe *reset*.
Parameter: tipe *reset*.
Kembalian: *object* ResetCommand.
- public ResetCommand setRef(String ref)
Mengatur nama *reference* tujuan dalam operasi *reset*. Secara default, *reference* yang digunakan adalah *HEAD*.
Parameter: nama dari *reference*.
Kembalian: *object* ResetCommand.
- public Ref call()
Berfungsi untuk menjalankan operasi *reset*.

2.3.10 Kelas ResetCommand.ResetType

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

- public static final ResetCommand.ResetType SOFT
Hanya mengubah posisi *HEAD*.
- public static final ResetCommand.ResetType HARD
Mengubah posisi *HEAD* ke *reference* tujuan. Selain itu, dilakukan *reset* pada *staging area* dan *working directory* sehingga perubahan yang terdapat pada *staging area* dan *working directory* akan hilang.
- public static final ResetCommand.ResetType MIXED
Mengubah posisi *HEAD* ke *reference* tujuan. Selain itu, dilakukan *reset* pada *staging area* sehingga perubahan yang terdapat pada *staging area* akan dipindahkan ke *working directory*.

2.3.11 Kelas RevCommit

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

- public final String getFullMessage()
Berfungsi untuk mendapatkan deskripsi dari suatu *commit*.
Kembalian: deskripsi dari suatu *commit* dalam bentuk String.
- public final String getShortMessage()
Berfungsi untuk mendapatkan deskripsi dari suatu *commit*. Jika panjang deskripsi *commit* lebih dari satu baris, maka hanya baris pertama yang dikembalikan.
Kembalian: baris pertama dari deskripsi *commit* dalam bentuk String.
- public final String getName()
Method ini mengembalikan *commit ID*.
Kembalian: *commit ID* berupa SHA-1 hash value dalam bentuk String.

- public final PersonIdent getAuthorIdent()

Berfungsi untuk mendapatkan informasi mengenai *author* yang melakukan *commit*.

Kembalian: objek *PersonIdent* yang memuat informasi tentang *author*(nama dan *email*) dan waktu dilakukannya *commit*.

2.3.12 Kelas PersonIdent

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

- public String getName()

Berfungsi untuk mengembalikan nama dari *author* yang melakukan *commit*.

Kembalian: nama dari *author*.

- public String getEmailAddress()

Berfungsi untuk mengembalikan alamat *email* dari *author* yang melakukan *commit*.

Kembalian: alamat *email* dari *author*.

- public Date getWhen()

Berfungsi mengembalikan waktu dilakukannya suatu *commit* oleh *author*.

Kembalian: waktu dilakukannya suatu *commit* berupa objek *Date*.

2.4 Selenium WebDriver

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

WebDriver merupakan kakas untuk mengotomatisasi pengujian pada aplikasi *web*[6]. WebDriver dapat berkomunikasi secara langsung dengan *browser* menggunakan *native support* pada *browser* tanpa melakukan injeksi kode JavaScript. Berikut ini adalah WebDriver yang terdapat pada Selenium WebDriver:

- ChromeDriver

Merupakan implementasi WebDriver yang mengontrol Chrome *browser*.

- FirefoxDriver

Merupakan implementasi WebDriver yang mengontrol Firefox *browser*.

- OperaDriver

Merupakan implementasi WebDriver yang mengontrol Opera *browser*.

- SafariDriver

Merupakan implementasi WebDriver yang mengontrol Safari *browser*.

- InternetExplorerDriver

Merupakan implementasi WebDriver yang mengontrol Internet Explorer *browser*.

- EdgeDriver

Merupakan implementasi WebDriver yang mengontrol Microsoft Edge *browser*.

Pada skripsi ini *tools* Selenium yang digunakan hanya Selenium WebDriver. Bahasa pemrograman yang digunakan adalah java. WebDriver yang digunakan adalah ChromeDriver. Pemilihan WebDriver ini hanya berdasarkan preferensi penulis saja. Penulis tidak memilih InternetExplorerDriver, EdgeDriver, dan SafariDriver karena InternetExplorerDriver dan EdgeDriver hanya bisa berjalan pada sistem operasi Windows, selain itu SafariDriver hanya bisa berjalan pada sistem operasi macOS. HtmlUnitDriver tidak dipilih karena WebDriver ini tidak mengimplementasikan *interface* untuk mengambil *screenshot*, sehingga tidak bisa dilakukan pengambilan *screenshot*. Pada subbab berikut dijelaskan beberapa kelas dan *interface* dari *library* Selenium WebDriver. Subbab ini mengacu pada [7].

2.4.1 Interface WebDriver

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

- void close()
Berfungsi untuk menutup *window* pada *browser*, jika *window* yang sekarang merupakan satu-satunya *window* yang terbuka maka *browser* akan ditutup.
- void quit()
Berfungsi untuk menutup *browser* dan semua *window* yang sedang terbuka.
- void get(String url)
Berfungsi untuk memuat halaman *web* pada *window* yang sedang aktif. *Method* ini mengirim *HTTP GET Request* untuk memuat halaman, dan *method* ini akan melakukan *blocking* sampai halaman *web* selesai dimuat.
Parameter: alamat *url* untuk memuat halaman *web*.
- String getTitle()
Berfungsi untuk mengembalikan judul dari halaman *web* yang sedang aktif.
Kembalian: judul dari halaman *web*.
- String getCurrentUrl()
Berfungsi untuk mendapatkan URL yang sedang aktif di *browser*.
Kembalian: URL dari halaman *web* yang sedang dimuat di *browser*.
- WebDriver.Options manage()
Berfungsi untuk mendapatkan interface WebDriver.Options.
Kembalian: interface dengan tipe WebDriver.Options.

2.4.2 Interface WebDriver.Options

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

- void deleteAllCookies()
Berfungsi untuk menghapus semua *cookie* yang terdapat pada *browser*.
- WebDriver.Window window()
Berfungsi untuk mendapatkan *interface* untuk mengatur *browser window*.
Kembalian: objek bertipe WebDriver.Window.

2.4.3 Interface WebDriver.Window

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

- void maximize()

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

2.4.4 Kelas ChromeDriver

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

- public ChromeDriver()

Constructor ini membuat instans dengan tipe ChromeDriver.

2.4.5 Kelas FirefoxDriver

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

- public FirefoxDriver()

Constructor ini membuat instans dengan tipe FirefoxDriver.

2.4.6 Kelas OperaDriver

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

- public OperaDriver()

Constructor ini membuat instans dengan tipe OperaDriver.

2.4.7 Kelas SafariDriver

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

- public SafariDriver()

Constructor ini membuat instans dengan tipe SafariDriver.

2.4.8 Kelas InternetExplorerDriver

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

- public InternetExplorerDriver()

Constructor ini membuat instans dengan tipe InternetExplorerDriver.

2.4.9 Kelas EdgeDriver

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

- public EdgeDriver()

Constructor ini membuat instans dengan tipe EdgeDriver.

2.4.10 Kelas HtmlUnitDriver

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

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

2.4.11 Interface OutputType

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

- static final OutputType<String> BASE64
Berfungsi untuk mendapatkan *screenshot* dalam bentuk *base64 data*.
- static final OutputType<byte[]> BYTES
Berfungsi untuk mendapatkan *screenshot* dalam bentuk *raw bytes*.
- static final OutputType<java.io.File> FILE
Berfungsi untuk mendapatkan *screenshot* dalam bentuk *temprorary file* yang akan dihapus setelah program keluar dari *Java Virtual Machine*.

2.4.12 Interface TakesScreenshot

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

- <X> X getScreenshotAs(OutputType<X> target) throws WebDriverException
Method ini berfungsi untuk mengambil *screenshot* dan mengembalikan hasilnya.
Kembalian: objek hasil *screenshot*
Parameter: tipe *output* yang diinginkan(lihat subbab [2.4.11](#)).

2.5 Apache Commons CLI

Library Apache Commons CLI menyediakan API untuk melakukan *parsing* argumen *command line* yang dikirimkan ke program^[8]. Apache Commons CLI termasuk ke dalam salah satu *project* Apache Commons. Tujuan utama dari *project* Apache Commons adalah membuat dan melakukan *maintain* pada komponen Java yang *reusable*. Pada subbab berikut dijelaskan beberapa kelas dan *interface* dari *library* Apache Commons CLI. Subbab ini mengacu pada ^[9].

2.5.1 Interface CommandLineParser

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

- CommandLine parse(Options options, String[] arguments) throws ParseException
Berfungsi untuk melakukan *parsing* argumen *command line* berdasarkan pada *option* yang telah ditentukan. *Method* ini melempar ParseException jika terjadi masalah saat melakukan *parsing*.

Parameter: *option* yang telah ditentukan, argumen *command line*.

Kembalian: objek CommandLine.

- `CommandLine parse(Options options, String[] arguments, boolean stopAtNonOption) throws ParseException`

Berfungsi untuk melakukan *parsing* argumen *command line* berdasarkan pada *option* yang telah ditentukan.

Parameter: *option* yang telah ditentukan, argumen *command line*, dan suatu *boolean* yang menentukan apakah *parsing* dihentikan jika terdapat *option* yang tidak valid. Jika bernilai *true*, *parsing* akan dihentikan dan semua argumen yang sudah diuraikan akan ditambahkan ke objek CommandLine. Jika bernilai *false*, akan dilempar ParseException bila terdapat *option* yang tidak valid.

Kembalian: objek CommandLine.

2.5.2 Kelas DefaultParser

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

- `CommandLine parse(Options options, String[] arguments) throws ParseException`

Berfungsi untuk melakukan *parsing* argumen *command line* berdasarkan pada *option* yang telah ditentukan. *Method* ini melempar ParseException jika terjadi masalah saat melakukan *parsing*.

Parameter: *option* yang telah ditentukan, argumen *command line*.

Kembalian: objek CommandLine.

- `CommandLine parse(Options options, String[] arguments, boolean stopAtNonOption) throws ParseException`

Berfungsi untuk melakukan *parsing* argumen *command line* berdasarkan pada *option* yang telah ditentukan.

Parameter: *option* yang telah ditentukan, argumen *command line*, dan suatu *boolean* yang menentukan apakah *parsing* dihentikan jika terdapat *option* yang tidak valid. Jika bernilai *true*, *parsing* akan dihentikan dan semua argumen yang sudah diuraikan akan ditambahkan ke objek CommandLine. Jika bernilai *false*, akan dilempar ParseException bila terdapat *option* yang tidak valid.

Kembalian: objek CommandLine.

2.5.3 Kelas CommandLine

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

- `public String getOptionValue(String opt)`

Berfungsi untuk mendapatkan nilai dari suatu *option* sesuai dengan namanya.

Parameter: nama dari *option*.

Kembalian: nilai dari suatu *option*. Jika *option* tidak ditemukan, akan mengembalikan *null*.

- `public String[] getOptionValues(Option option)`

Berfungsi untuk mendapatkan kumpulan nilai dari suatu *option*. Parameter: nama dari *option*.

Kembalian: kumpulan nilai dari suatu *option* dengan tipe *array of String*. Jika *option* tidak ditemukan, akan mengembalikan *null*.

- `public Option[] getOptions()`

Berfungsi untuk mengembalikan *option* hasil *parsing*.

Kembalian: *array* dari *option* hasil *parsing*.

2.5.4 Kelas Options

Kelas ini merepresentasikan kumpulan dari objek Option. Berikut ini adalah *method* yang dimiliki kelas ini:

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

2.5.5 Kelas Option

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

- public String getLongOpt()
Berfungsi untuk mendapatkan nama panjang dari suatu *option*.
Kembalian: nama panjang dari suatu *option*.
- public String getValue()
Berfungsi untuk mendapatkan nilai dari *option*.
Kembalian: nilai dari *option*. Jika terdapat lebih dari satu kembalian, hanya nilai pertama yang dikembalikan. Jika tidak ada nilai, akan mengembalikan *null*.
- public String[] getValues()
Berfungsi untuk mendapatkan nilai-nilai dari *option*.
Kembalian: nilai-nilai dari *option* berupa *array of String*. Jika tidak ada nilai, akan mengembalikan *null*.
- public boolean hasArg()
Berfungsi untuk mengetahui apakah suatu *option* membutuhkan argumen.
Kembalian: *true* jika *option* ini membutuhkan argumen , *false* jika *option* ini tidak membutuhkan argumen.
- public boolean hasArg()
Berfungsi untuk mengetahui apakah suatu *option* dapat menerima lebih dari satu argumen.
Kembalian: *true* jika *option* ini dapat menerima lebih dari satu argumen , *false* jika tidak.
- public String getDescription()
Berfungsi untuk mendapatkan deskripsi dari suatu *option*.
Kembalian: deskripsi dari *option* ini.
- public String getArgName()
Berfungsi untuk mendapatkan nama dari suatu *option*.
Kembalian: nama dari argumen suatu *option*

2.5.6 Kelas Option.Builder

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

- public Option.Builder desc(String description)
Berfungsi untuk memberikan deskripsi pada *option*.
Parameter: deskripsi dari *option*.
Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.

- public Option.Builder longOpt(String longOpt)
Berfungsi untuk memberikan nama panjang pada *option*.
Parameter: nama panjang *option*.
Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- public Option.Builder hasArg()
Berfungsi untuk menyatakan bahwa *option* ini membutuhkan argumen.
Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- public Option.Builder hasArgs()
Berfungsi untuk menyatakan bahwa *option* ini membutuhkan argumen dimana jumlah argumen bisa lebih dari satu.
Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- public Option.Builder argName(String argName)
Berfungsi untuk memberi nama pada argumen.
Parameter: nama argumen.
Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- public Option.Builder required(boolean required)
Berfungsi untuk menyatakan bahwa *option* ini wajib ada.
Parameter: variabel bertipe *boolean* yang menentukan apakah *option* ini wajib ada.
Kembalian: objek Option.Builder yang bisa digunakan untuk *method chaining*.
- public Option build()
Berfungsi untuk membuat objek Option berdasarkan nilai pada kelas Option.Builder.
Kembalian: objek Option.

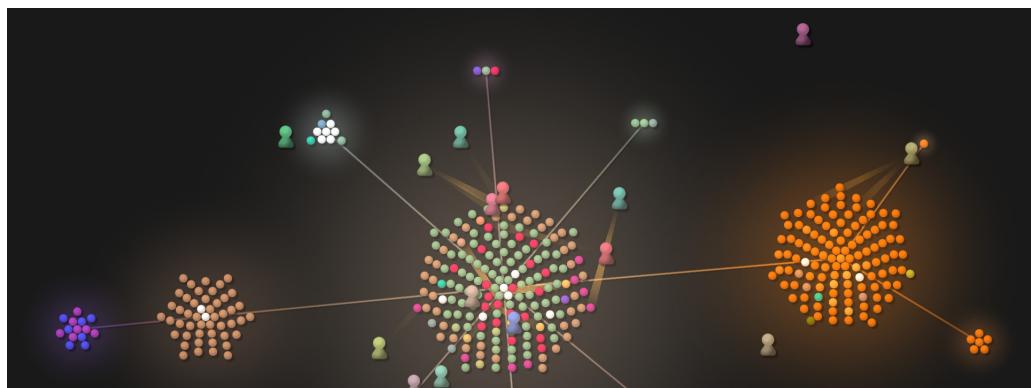
BAB 3

ANALISIS

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

3.1 Analisis Aplikasi Sejenis

Aplikasi sejenis yang digunakan untuk membangkitkan animasi adalah Gource. Reposisori perangkat lunak ditampilkan oleh Gource sebagai animasi pohon dimana pusatnya adalah *root directory* dari repositori perangkat lunak[10]. Direktori ditampilkan sebagai *branch*, sedangkan *file* ditampilkan sebagai *leaf*. Developer dapat terlihat di *working directory* pada saat mereka berkontribusi untuk repositori.



Gambar 3.1: Visualisasi dari repositori perangkat lunak menggunakan Gource.

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

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

1. `gource -<WIDTH>x<HEIGHT>`
Opsi ini berfungsi untuk mengatur resolusi layar dari animasi. Parameter dari opsi ini adalah lebar dan panjang layar dalam satuan piksel.
2. `gource -camera-mode <MODE>`
Opsi ini berfungsi untuk mengatur mode kamera pada Gource. Parameter dari opsi ini adalah mode dari kamera. Terdapat dua mode yaitu *overview* dan *track*. Dalam mode *track*, kamera

bergerak mengikuti *user* yang sedang aktif. Dalam mode *overview*, kamera menampilkan seluruh repositori.

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

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

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

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

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

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

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

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

7. **gource -disable-bloom**

Opsi ini berfungsi untuk menonaktifkan animasi *bloom*.

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

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

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

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

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

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

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

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

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

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

13. **gource -logo <IMAGE>**

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

14. **gource -logo-offset <X>x<Y>**

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

15. **gource -title <TITLE>**

Opsi ini berfungsi untuk memberi judul. Judul tersebut ditampilkan pada pojok kiri bawah layar.

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

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

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

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

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

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

- *bloom*: efek *bloom*.
- *date*: waktu.
- *dirnames*: nama direktori.
- *files*: ikon dari berkas.
- *filenames*: nama berkas.
- *root*: *root directory*.
- *users*: ikon dari *user*.
- *usernames*: nama dari *user*.

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

Gource dapat digunakan untuk berbagai macam proyek perangkat lunak. Program pada skripsi ini berfokus pada proyek perangkat lunak berbasis *web*. Tidak seperti Gource yang menampilkan direktori dan *file* pada animasi, program pada skripsi ini menampilkan *screenshot* halaman *web* pada animasi.

3.2 Analisis Penggunaan JGit dan Selenium WebDriver

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

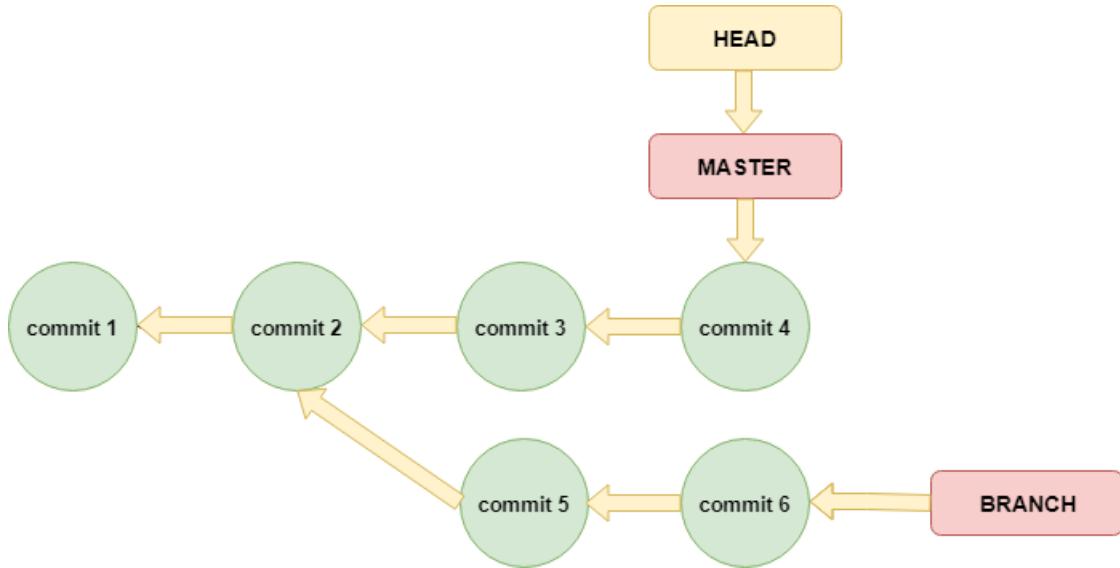
3.2.1 Analisis Penggunaan JGit

JGit dapat digunakan untuk berinteraksi dengan repositori yang terekam oleh Git. 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. *Constructor* FileRepository menerima parameter berupa alamat dari direktori Git.

Dibutuhkan beberapa langkah untuk melakukan operasi Git Log. Dari object bertipe Git, bisa didapatkan *object* bertipe LogCommand dengan memanggil *method* log(). Setelah itu, *method* call() dipanggil 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 commit history seperti pada Gambar 3.2 dan saat ini

HEAD menunjuk ke Master, *Iterable<RevCommit>* akan berisi commit4, commit3, commit2, dan commit1. Jika saat ini *HEAD* menunjuk ke BRANCH maka *Iterable<RevCommit>* akan berisi commit6, commit5, commit2, dan commit1. Operasi Git Log ini dipakai oleh program dalam skripsi ini untuk mendapatkan seluruh *commit history* dari repositori perangkat lunak berbasis *web*.



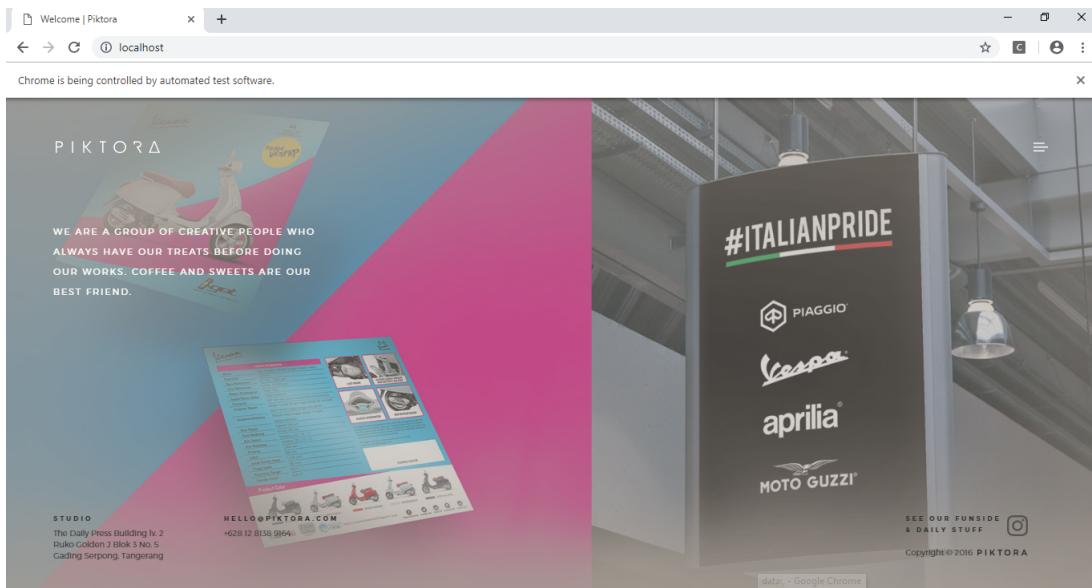
Gambar 3.2: *Commit history* direpresentasikan sebagai Directed Acyclic Graph.

Dibutuhkan beberapa langkah untuk melakukan operasi Git Checkout. Dari object bertipe Git, bisa didapatkan *object* bertipe CheckoutCommand dengan memanggil *method* checkout(). Setelah itu, *method* setName() dipanggil dengan parameter berupa *commit ID*. Kemudian *method* call() dipanggil untuk menjalankan operasi Git Checkout. Setelah operasi Git Checkout dijalankan, *working directory* akan diperbarui sesuai dengan *state* pada *commit* tertentu. Operasi *checkout* ini digunakan oleh program dalam skripsi ini untuk menelusuri *commit history* dan mendapatkan halaman *web* dari repositori perangkat lunak berbasis *web*.

3.2.2 Analisis Penggunaan Selenium WebDriver

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

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



Gambar 3.3: *Browser* yang dikontrol oleh ChromeDriver.

TakesScreenshot merupakan *interface* yang digunakan untuk menangkap *screenshot* halaman *web* pada WebDriver. Kelas ChromeDriver mengimplementasikan *interface* ini. *Method* yang digunakan untuk menangkap *screenshot* adalah getScreenshotAs(), diikuti dengan parameter bertipe OutputType. OutputType ini merupakan *interface* yang digunakan untuk menentukan tipe *output* dari hasil *screenshot*. OutputType yang digunakan pada program dalam skripsi ini adalah File. Tipe ini dipilih karena memudahkan kelas ImageIO¹ dalam membaca *image*. Pada OutputType BYTES, byte harus dibaca dulu oleh kelas ByteArrayInputStreamStream², kemudian dibaca oleh kelas ImageIO. Pada OutputType BASE64, base64 *data* dikonversi menjadi *array of bytes*, setelah itu dibaca oleh ByteArrayInputStreamStream, kemudian dibaca oleh kelas ImageIO. Pada OutputType FILE, *file* dapat langsung dibaca oleh kelas ImageIO. *Interface* TakesScreenshot ini digunakan oleh program dalam skripsi ini untuk mengambil *screenshot* dari halaman *web* pada repositori perangkat lunak berbasis *web*.

3.3 Analisis Fitur Aplikasi yang Dibangun

Pada skripsi ini, dibuat sebuah perangkat lunak yang dapat membangun animasi *timelapse* dari pengembangan proyek perangkat lunak berbasis web. Yang dibuat animasinya adalah halaman web dari perangkat lunak. Jumlah halaman berkisar antara satu sampai empat halaman, tergantung pada masukan dari *user*. *Output* dari program adalah *file* hasil animasi yang bertipe GIF. Program menerima masukan dari argumen *command line*. Konfigurasi program diatur melalui Command Line Option.

Berikut ini adalah *option* yang terdapat pada program:

1. **-path <PATH>**

Opsi ini berfungsi untuk mengatur *path* dari repositori perangkat lunak berbasis *web* yang akan dibuat animasinya. Parameter dari opsi ini adalah *path* dari repositori perangkat lunak berbasis *web* yang terekam oleh Git. Opsi ini mengacu pada opsi **-path** yang terdapat pada Gource. Opsi ini wajib ada.

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

Opsi ini berfungsi untuk menjalankan *terminal command*. *Terminal command* ini dijalankan

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

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

sebelum mengambil *screenshot*. Parameter dari opsi ini adalah *terminal command* dalam bentuk *string* dan harus diapit dengan tanda petik ganda. Opsi ini secara khusus dibuat karena program dalam skripsi ini membuat animasi *timelapse* pada suatu repositori Git yang berisi perangkat lunak berbasis *web*. Perangkat lunak berbasis *web* tersebut dijalankan di komputer lokal dan terkadang membutuhkan *setup* tambahan dimana *setup* tersebut bisa diatur menggunakan opsi ini. Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, program tidak menjalankan *terminal command*.

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

Opsi ini berfungsi untuk mengatur alamat *url* dari halaman *web* dimana dilakukan pengambilan *screenshot* pada halaman ini. Jumlah halaman bisa lebih dari satu, dengan jumlah maksimal empat halaman. Parameter dari opsi ini adalah alamat *url* dari halaman *web*, dengan jumlah maksimal empat alamat *url*. Opsi ini secara khusus dibuat karena program dalam skripsi ini membuat animasi *timelapse* dari kumpulan *screenshot* halaman *web* dimana *screenshot* tersebut didapatkan dari suatu alamat *url*. Opsi ini wajib ada.

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

Opsi ini berfungsi untuk mengatur durasi munculnya satu *commit* pada animasi. Opsi ini berfungsi juga untuk mengatur kecepatan animasi. Parameter dari opsi ini adalah durasi munculnya satu *commit* dalam satuan detik. Parameter harus berupa bilangan bulat atau riil dimana nilainya lebih besar dari nol dan kurang dari 656. Opsi ini mengacu pada opsi **-seconds-per-day** yang terdapat pada Gource dengan sedikit penyesuaian. Program pada skripsi ini berfokus pada pembuatan animasi berdasarkan pada *commit*, sehingga kecepatan animasinya diatur berdasarkan pada durasi kemunculan *commit*. Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, nilai defaultnya adalah 1 detik.

5. **-title <TITLE>**

Opsi ini berfungsi untuk memberi judul. Judul tersebut ditampilkan pada pojok kiri bawah layar. Opsi ini mengacu pada opsi **-title** yang terdapat pada Gource. Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, maka judul tidak ditampilkan.

6. **-logo <IMAGE>**

Opsi ini berfungsi untuk memasukkan logo. Logo tersebut ditampilkan pada pojok kanan bawah layar. Parameternya adalah *path* dari *file* gambar. Format gambar yang dapat diterima antara lain JPG, PNG, GIF dan, BMP. Opsi ini mengacu pada opsi **-logo** yang terdapat pada Gource. Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, maka logo tidak ditampilkan.

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

Opsi ini berfungsi untuk mengatur *commit* awal untuk membangkitkan animasi. Parameter dari opsi ini adalah *commit ID* awal dimana panjang *commit ID* adalah tujuh sampai sepuluh karakter. Opsi ini mengacu pada opsi **-start-date** yang terdapat pada Gource dengan sedikit penyesuaian. Program pada skripsi ini berfokus pada pembuatan animasi berdasarkan pada *commit*, sehingga rentang animasi diatur dengan menggunakan rentang *commit*. Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, program membangkitkan animasi dimulai dari *commit* paling awal.

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

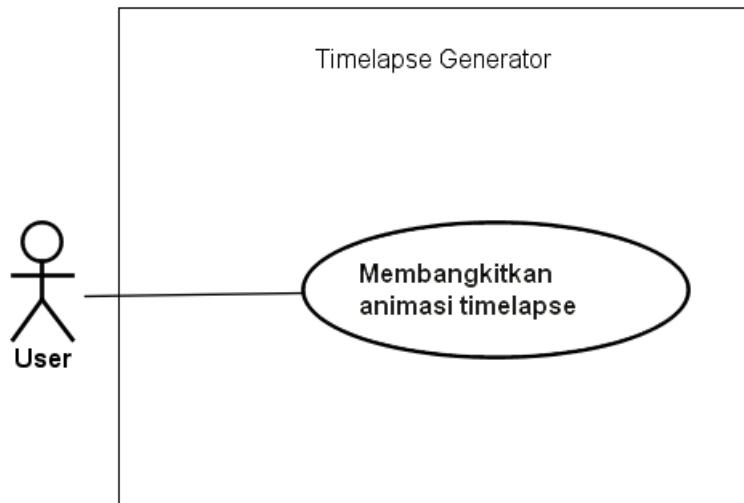
Opsi ini berfungsi untuk mengatur *commit* akhir untuk membangkitkan animasi. Parameter dari opsi ini adalah *commit ID* akhir dimana panjang *commit ID* adalah tujuh sampai sepuluh karakter. Opsi ini mengacu pada opsi **-stop-date** yang terdapat pada Gource dengan sedikit penyesuaian. Program pada skripsi ini berfokus pada pembuatan animasi berdasarkan pada *commit*, sehingga rentang animasi diatur dengan menggunakan rentang *commit*. Opsi ini

bersifat opsional. Jika opsi ini tidak dimasukkan, program membangkitkan animasi sampai dengan *commit* paling akhir.

9. `-branch <BRANCH>`

Opsi ini berfungsi untuk mengatur *branch* dari repositori perangkat lunak berbasis *web* yang akan dibangkitkan animasinya. Parameter dari opsi ini adalah nama dari *branch*. Opsi ini secara khusus dibuat karena program dalam skripsi ini membuat animasi *timelapse* pada suatu repositori Git, sedangkan Gource tidak hanya membuat visualisasi dari repositori Git tetapi juga dari VCS lainnya. Selain itu kode program pada repositori Git belum tentu berada pada *branch* master, bisa saja berada pada *branch* lain. Opsi ini bersifat opsional. Jika opsi ini tidak dimasukkan, program membangkitkan animasi pada *branch* master.

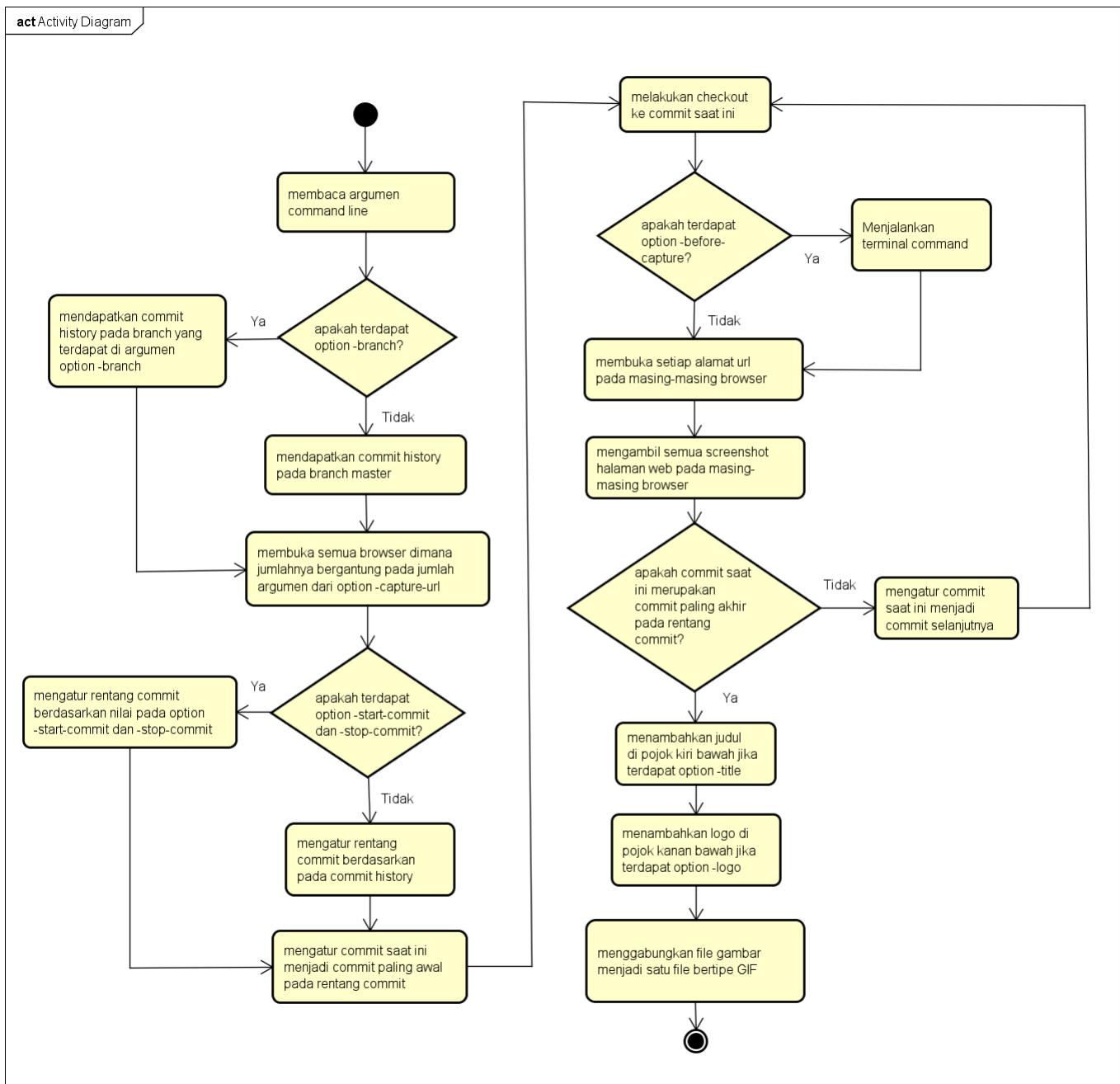
Program pada skripsi ini memiliki satu fitur. Fitur tersebut adalah membangkitkan animasi *timelapse*. Penjelasan fitur dapat dilihat pada Gambar 3.4 dan Tabel 3.1.



Gambar 3.4: *Use case diagram* perangkat lunak.

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"> 1. Sistem membaca <i>input</i> dari argumen <i>command line</i>. 2. Sistem menelusuri sejarah pengembangan perangkat lunak berbasis <i>web</i> dengan bantuan JGit. Saat menelusuri sejarah pengembangan perangkat lunak, sistem mengambil <i>screenshot</i> dari halaman <i>web</i> menggunakan SeleniumWebDriver. 3. Sistem menggabungkan <i>screenshot</i> menjadi satu file bertipe GIF.

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



Gambar 3.5: Langkah-langkah untuk membangkitkan animasi *timelapse*.

Gambar 3.5 menunjukkan langkah-langkah untuk membangkitkan animasi *timelapse*. Penggabungan *screenshot* halaman-halaman *web* menjadi *file* *GIF* dilakukan dengan bantuan *library*³.

3.4 Prapengujian Website Piktora

Pengujian dilakukan dengan proyek Piktora sebagai input dari program. Piktora⁴ merupakan situs *web* yang menawarkan layanan *creative design* dan *branding*. Layanan yang ditawarkan berupa *graphic design* untuk poster, *banner*, *website*, dan aplikasi *mobile*. Repositori dari situs *web* ini

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

⁴<http://www.piktora.com>

disimpan pada Gitlab⁵, dibutuhkan akses khusus untuk membuka repositori tersebut. Piktora dibangun dengan php menggunakan framework Code Igniter. Piktora menggunakan MySQL sebagai basis datanya.

Prapengujian dilakukan menggunakan prototipe program yang dilengkapi pada Bab 5. Berikut adalah spesifikasi perangkat keras dan perangkat lunak yang digunakan untuk prapengujian:

- Processor: Intel Core i3 4030U
- RAM: 6GB
- Sistem Operasi: Windows 10 pro 64-bit
- Apache HTTP Server versi 2.4.29
- MySQL Server versi 5.5.5
- Netbeans versi 8.1
- Google Chrome versi 73.0.3683.86

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

Repositori Piktora memiliki 58 *commit*. Listing 3.1 menunjukkan *commit history* pada repositori Piktora, *commit history* didapatkan dengan menjalankan operasi Git Log pada terminal. Setelah dilakukan pengujian terdapat beberapa masalah. Masalah tersebut yaitu perbedaan letak *file*, migrasi *database*, dan konfigurasi *database*. Masalah-masalah tersebut dibahas pada subbab 3.4.1 sampai 3.4.3.

Listing 3.1: *Commit history* pada proyek Piktora

```

1 315d374 - Oct 31 16:52:46 2016 : Basic CI files + htaccess & webconfig + database.
2      php ignore
3
4 27ce3d4 - Nov 5 13:12:43 2016 : setup environment for piktora
5
6 65f0c9c - Nov 5 19:22:58 2016 : * create structure for all pages * add dummy
7      images
8
9 bffbae1 - Nov 8 18:00:32 2016 : - basic structure (navbar semi complete) - add
10     fonts
11
12 5c59916 - Nov 8 19:51:18 2016 : implement navbar, footer, and projects/ page
13
14 7738380 - Nov 8 20:05:27 2016 : fix pc and ipad navbar fontsize
15
16 26bdbbe - Nov 8 20:16:33 2016 : fix position image for desktop /projects
17
18 3db3ce8 - Nov 9 00:28:27 2016 : - implements project details page - fix some minor
19     issue - add some project image
20
21 5ef34fa - Nov 13 13:01:06 2016 : implement about us (raw version)
22
23 3caf535 - Nov 15 11:55:15 2016 : fix minor issues view/about: - background-image
24      position. Make it to the center position - slick.js img need to set to inline
25
26 c5eb3b6 - Nov 15 13:02:42 2016 : implement /welcome page
27
28 ade9216 - Nov 15 13:12:08 2016 : fix minor issues: - move style footer to global -
29      add space before PIKTORA
30
31
32 c77b5b3 - Nov 18 18:18:25 2016 : implement /contact
33
34 b42b819 - Nov 18 21:22:22 2016 : change a href to style cursor: pointer
35
36 3eb7af8 - Nov 21 16:09:40 2016 : .htaccess to be compatible with cloud kilat
37
38 e87e84b - Nov 22 14:53:45 2016 : - change vw to 100% - add captcha
39
40 ff8d829 - Nov 22 15:22:53 2016 : Solved captcha font load: use otf instead of ttf
41
42 Also: create directory assets/img/captcha and ignore everything inside

```

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

```

24
25 dc87342 - Nov 22 15:49:03 2016 : - implement captcha code - remove wrong css
26 9ebe433 - Nov 22 16:12:44 2016 : add scroll feature in project/
27 f0f7270 - Nov 23 15:17:42 2016 : Added Google PHP Client v2 See https://github.com
   /google/google-api-php-client
28
29 57a239b - Nov 23 15:19:57 2016 : Merge origin/master
30 e2dfebe - Nov 27 14:42:24 2016 : - remove blue outline when click with slick -
   change background image in about to newest one
31
32 a19e7f2 - Nov 27 15:24:42 2016 : Detailing dari Edina: 1. Hal. Project Detail ,
   font coba diperkecil saja mungkin ya. 3. Beberapa ukuran font dan spacing ada
   yang kurang pas sedikit , terlampir detail revisinya ya (file pdf) 4. Footer
   dibuat selalu stay terus di bagian bawah dengan posisi yang selalu sama. Di home
   & contact sudah sama, namun di hal. product posisinya agak lebih naik.
33
34 3d79d0a - Nov 27 15:29:58 2016 : fix minor issue
35 0fc958 - Nov 28 10:11:06 2016 : add raw admin contents
36 add3974 - Nov 28 12:03:21 2016 : add summernote, implement read project
37 0680488 - Nov 29 12:38:23 2016 : update admin for projects
38 fbe7639 - Nov 29 13:10:30 2016 : implement admin for home
39 db0cedd - Nov 29 13:38:29 2016 : - implementasi database bagian user - upload 9
   gambar contoh project
40
41 0fe9aaf - Nov 29 14:17:20 2016 : ubah warna garis captcha
42 f2326dd - Nov 29 14:44:51 2016 : - lewati proses otentikasi sementara
43 f78cdb4 - Dec 2 12:10:47 2016 : (trying to) fix issue #2
44 ef9b62b - Dec 2 17:09:58 2016 : revisi dari edina ke-2
45 c689aa8 - Dec 2 17:11:13 2016 : perbaikan admin sedikit
46 c4e9576 - Dec 2 17:14:06 2016 : perbaikan di /contact , kelewat
47 02d04f1 - Dec 5 14:55:20 2016 : tambah wording
48 a4e4858 - Dec 5 15:08:59 2016 : perbaikan kata2 sedikit
49 bbd82c2 - Dec 6 10:41:40 2016 : implementasi email
50 f8c64fc - Dec 6 11:03:08 2016 : change to httpdocs
51 eb49c2b - Dec 6 11:35:20 2016 : hapus migrasi script di admin
52 ace1988 - Dec 6 11:39:00 2016 : change overflow to auto
53 627e65b - Dec 6 11:45:26 2016 : modify database back to local
54 0896f81 - Dec 7 16:08:30 2016 : update home versi mobile jadi baru (revisi dari
   Edina)
55
56 5cf1292 - Dec 7 16:21:01 2016 : ubah background di about menjadi tidak pecah
57 c83f4aa - Thu Dec 15 15:04:30 2016 : remove piktoria secrets
58 57f5ea4 - Thu Dec 15 15:09:43 2016 : remove unimportant data
59 7931c21 - Dec 24 18:40:41 2016 : edit wording
60 9b0a302 - Dec 25 06:03:50 2016 : Another wording fix
61 f1ea410 - Thu Jan 5 15:23:32 2017 : fix instagram link
62 1880a88 - Thu Jan 5 15:24:12 2017 : Merge branch 'master' of https://github.com/
   pascalalfadian/Piktora
63
64 286aa78 - Jan 16 12:48:45 2017 : Perbaikan wording di admin edit project
65 33702c2 - Feb 21 13:31:08 2017 : change email sender to piktoria@mailgun.dnartworks.
   com.au
66
67 18c39ef - Thu Apr 13 15:21:49 2017 : Test commit (in gitlab). Nothing much important
68 9bfde3c - Apr 17 15:09:54 2017 : add ignore sftp-config.json
69 38711f0 - Apr 17 15:15:03 2017 : fix bug ugly display when projects too high
70 9f041ef - May 15 10:40:16 2017 : set insta url to https://www.instagram.com/
   piktorastudio/
71
72 6a085c1 - Dec 12 14:38:38 2017 : Update company address
73 89000be - Jan 12 12:25:30 2018 : Update new company address

```

3.4.1 Perbedaan Letak File

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

Solusi untuk mengatasi masalah ini adalah dengan menambahkan *option* yang menjalankan *terminal command*. *Terminal command* ini berisi perintah untuk menjalankan *script* PHP. *Script* PHP ini mengecek letak *file* "index.php" pada *folder* "www" dan "htdocs". *Script* kemudian mengecek *directory root* Apache pada *file* "httpd.conf". Jika *directory root* sudah mengarah ke *folder* tempat "index.php" berada, maka *script* tidak mengubah isi *file* "httpd.conf". Jika *root directory* tidak mengarah ke *folder* tempat "index.php" berada, maka *script* akan mengubah *root directory* pada *file* "httpd.conf" dan melakukan *restart* pada Apache.

3.4.2 Permasalahan Konfigurasi Database

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

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

```

1 $active_group = 'default';
2 $query_builder = TRUE;
3
4 $db[ 'default' ] = array(
5     'dsn' => '',
6     'hostname' => 'localhost',
7     'username' => 'piktora',
8     'password' => 'dmHx64%6',
9     'database' => 'piktora',
10    'dbdriver' => 'mysqli',
11    'dbprefix' => '',
12    'pconnect' => FALSE,
13    'db_debug' => (ENVIRONMENT !== 'production'),
14    'cache_on' => FALSE,
15    'cachedir' => '',
16    'char_set' => 'utf8',
17    'dbcollat' => 'utf8_general_ci',
18    'swap_pre' => '',
19    'encrypt' => FALSE,
20    'compress' => FALSE,
21    'stricton' => FALSE,
22    'failover' => array(),
23    'save_queries' => TRUE
24 );

```

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 MySQL Server 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 menjalankan *terminal command*. *Terminal command* ini berisi perintah untuk menjalankan *script PHP*. *Script* ini mengecek *password* yang terdapat pada *file "database.php"*. Jika tidak ditemukan *password* atau ditemukan *password* berupa *piktora*, maka *script* tidak mengubah isi *file "database.php"*. Jika ditemukan *password* berupa *dmHx64%6*, maka *script* mengubah *password* menjadi *piktora*.

3.4.3 Permasalahan Migrasi *Database*

Pada *commit* 0fcfd958 (28 November 2016), terjadi *error* saat melakukan migrasi database. Pada *commit* 3d79d0a (27 November 2016), terdapat satu *file* untuk melakukan migrasi yaitu "20161122150000_Structure.php". Pada *commit* 0fcfd958 (28 November 2016) terdapat dua *file* untuk melakukan migrasi yaitu "20161122150000_Structure.php" dan "20161122150001_InitialData.php". Pada *commit* 3d79d0a (27 November 2016) *file* "20161122150000_Structure.php" dijalankan saat melakukan migrasi. Versi migrasi *database* menjadi "20161122150000". Pada *commit* 0fcfd958 (28 November 2016), *file* "20161122150000_Structure.php" tidak dijalankan karena dianggap sama dengan versi migrasi *database* saat ini. Hanya *file* "20161122150001_InitialData.php" yang dijalankan pada *commit* 0fcfd958 (28 November 2016). Isi *file* "20161122150000_Structure.php" pada *commit* 3d79d0a (27 November 2016) dan 0fcfd958 (28 November 2016) berbeda. Hal ini yang menyebabkan terjadinya *error* saat melakukan migrasi database.

Sama seperti subbab 3.4.1, solusi untuk mengatasi masalah ini adalah dengan menambahkan *option* yang menjalankan *terminal command*. *Terminal command* ini berisi perintah untuk menjalankan *script PHP*. *Script* ini melakukan dua pekerjaan. Pertama, *script* menghapus *database* *piktora*. Setelah itu, *script* membuat dan melakukan migrasi basis data.

BAB 4

PERANCANGAN

Pada bab ini dijelaskan mengenai perancangan perangkat lunak yang dibangun, meliputi perancangan kelas dan perancangan antarmuka.

4.1 Perancangan Kelas

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

- BrowserController

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

- private final WebDriver[] drivers
Atribut ini adalah kumpulan WebDriver yang digunakan untuk mengontrol *browser*.
- 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.

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 seukuran layar.
- 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.

- public List<File> getScreenshotFiles()

Berfungsi untuk mendapatkan hasil *screenshot*.

- CommandLineOptions

Kelas ini berfungsi untuk menyimpan semua *option* yang terdapat dalam program ini, dan melakukan *parsing* argumen *command line* 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* 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* yang didapatkan dari kelas Main.
- public Properties getParsedOptions()

Berfungsi untuk mengembalikan *option* yang sudah diparsing.

- VCS

Kelas ini digunakan untuk berinteraksi dengan repositori perangkat lunak berbasis *web* yang terekam oleh Git. Berikut adalah atribut yang terdapat pada kelas ini:

- private final Git git

Atribut ini digunakan untuk melakukan interaksi pada repositori perangkat lunak yang terekam oleh Git.
- private final List<String> commitIDs

Atribut ini digunakan untuk menampung seluruh *commit ID* dari hasil penelusuran *commit history*.

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

- public VCS(String path, String branch)

Constructor dari kelas ini. Berfungsi untuk menginisialisasi variabel git dan mendapatkan seluruh *commit history* dari repositori perangkat lunak berbasis *web* pada *branch* tertentu. *Branch* tersebut diambil dari parameter constructor. Parameternya adalah *path* dari repositori perangkat lunak berbasis web dan nama *branch* yang digunakan untuk membangkitkan animasi.
- public void checkoutCommit(int commitIndex)

Berfungsi untuk melakukan *checkout* ke *commit* tertentu. Parameter dari *method* ini adalah indeks dari variabel commitIDs.
- public void checkoutMaster()

Berfungsi untuk melakukan *checkout* ke *commit* terakhir yang ada di *branch* master.
- public void hardReset()

Berfungsi untuk melakukan operasi Git Reset dengan tipe *hard reset*. Operasi ini menghapus perubahan pada *working directory* dan *staging area* di *commit* tertentu.
- public int getNumberOfCommit()

Berfungsi untuk mendapatkan jumlah *commit*.
- public int getCommitIndex(String commitID)

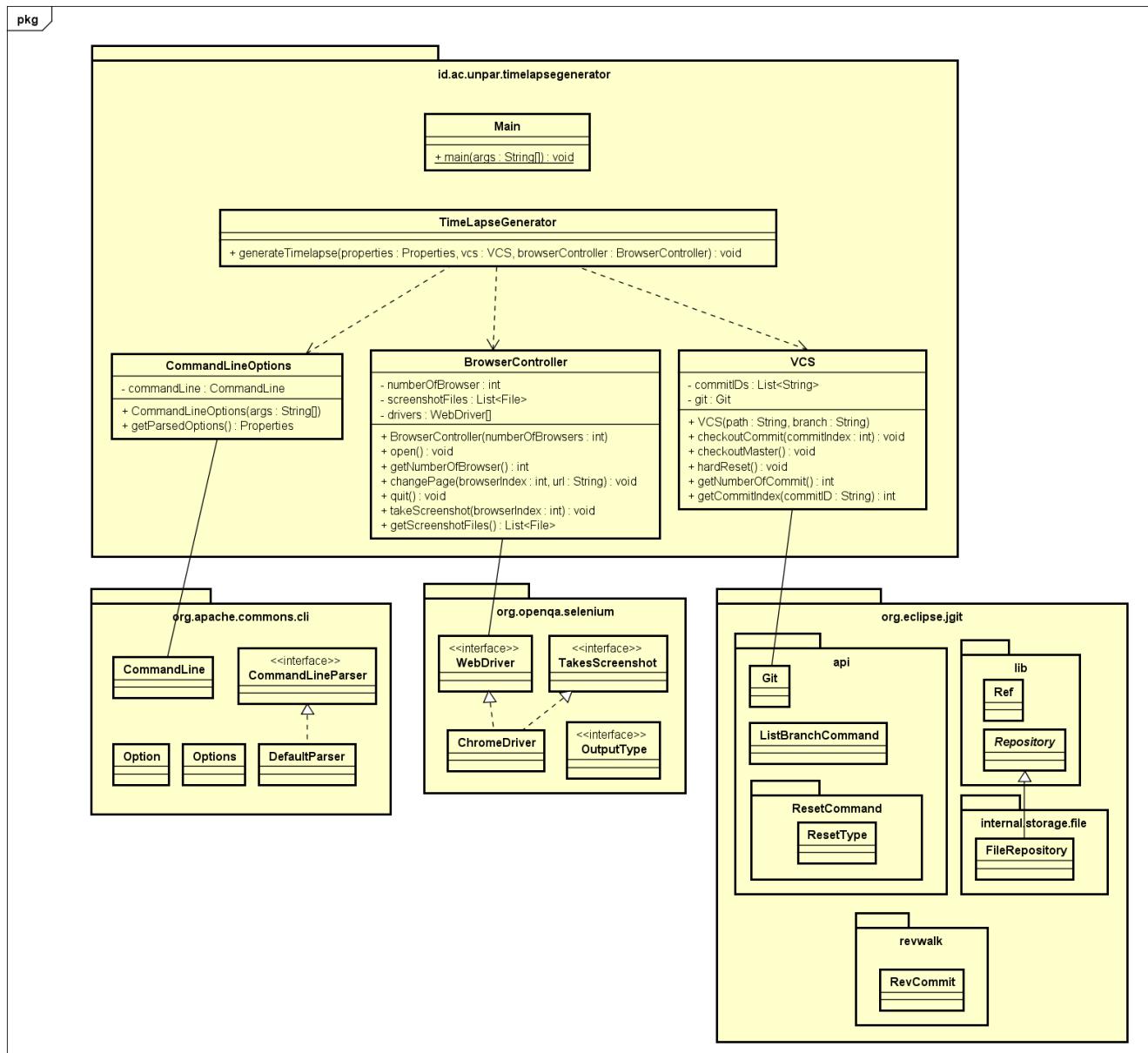
Berfungsi untuk mendapatkan indeks dari variabel commitIDs. Parameternya adalah *Commit ID* yang dicari indeksnya.

- TimeLapseGenerator

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

- public void generateTimelapse(Properties properties, VCS vcs, BrowserController browserController)

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



Gambar 4.1: Diagram kelas.

4.2 Perancangan Antarmuka

Program dalam skripsi ini menggunakan terminal sebagai antarmuka (Command Line Interface). *Input* dari program ini dimasukkan melalui argumen *command line*. *Option* yang dapat dimasukkan ke program ini dapat dilihat pada subbab 3.3. *Output* dari program ini berupa status pada terminal dan *file* hasil animasi bertipe GIF.

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

```
1 | Animasi timelapse berhasil dibuat
```

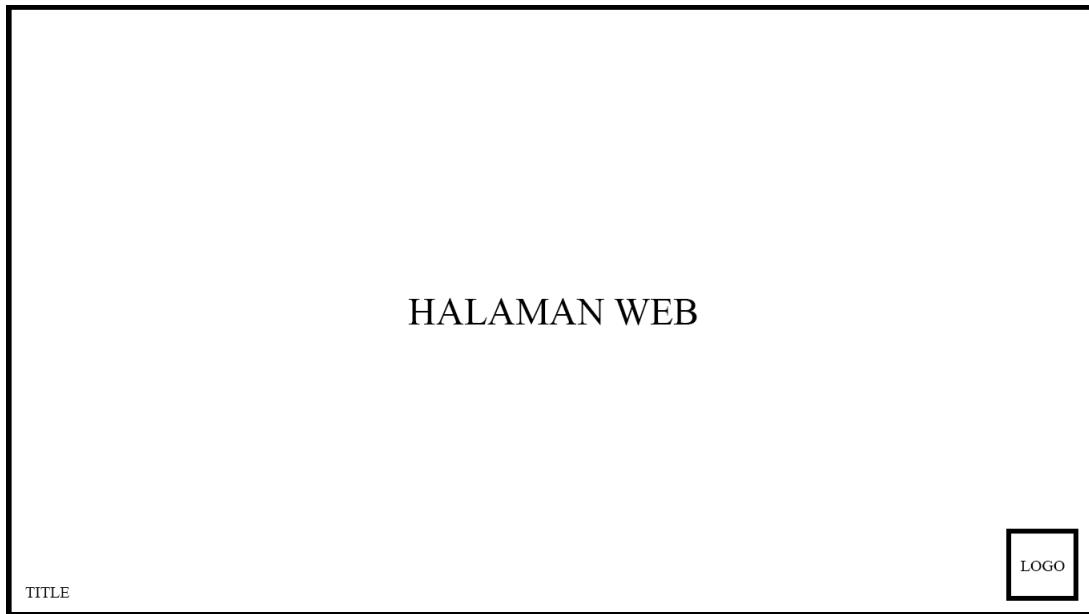
Listing 4.2: Status pesan yang muncul pada terminal saat program gagal membangkitkan animasi *timelapse*.

```
1 | Animasi timelapse gagal dibuat
2 | <ERROR MESSAGE>
```

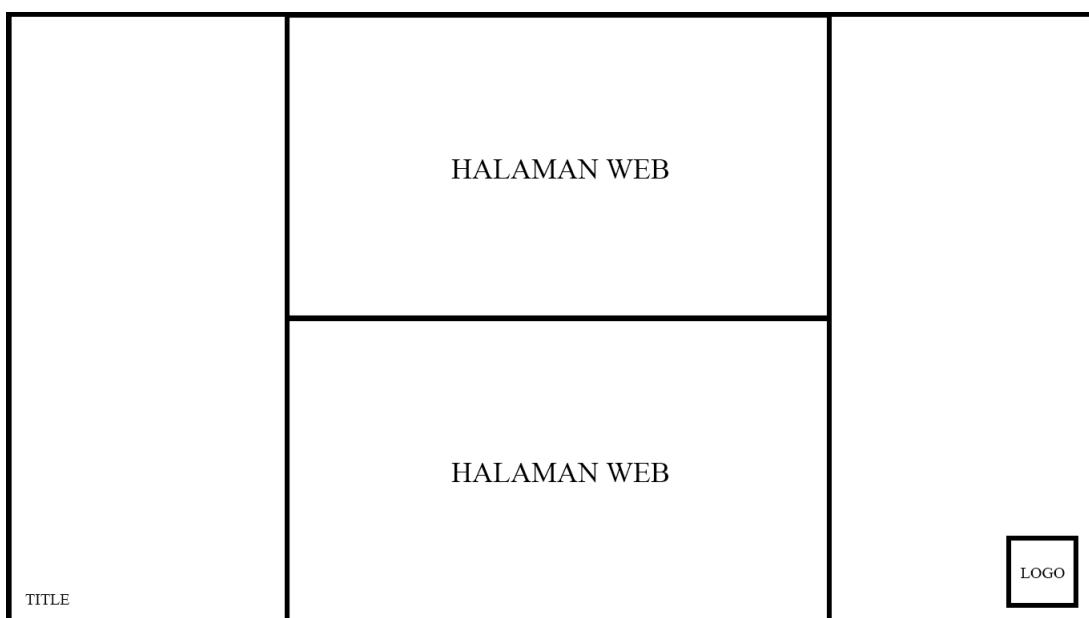
Listing 4.1 menunjukkan status yang ditampilkan pada terminal saat program berhasil membangkitkan animasi *timelapse*. Listing 4.2 menunjukkan status yang ditampilkan pada terminal saat program gagal membangkitkan animasi *timelapse*. Pada Listing 4.2, baris pertama menunjukkan bahwa program gagal membangkitkan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

- **Path repositori tidak valid**
Pesan ini muncul jika *path* repositori tidak valid.
- **Jumlah url yang dicapture maksimal 4**
Pesan ini muncul jika jumlah argumen pada *option capture-url* lebih dari 4.
- **Seconds per commit harus lebih besar dari 0**
Pesan ini muncul jika argumen dari *seconds-per-commit* bernilai lebih kecil dari 0.
- **Seconds per commit harus kurang dari sama dengan 655**
Pesan ini muncul jika argumen dari *seconds-per-commit* bernilai lebih besar dari 655.
- **Seconds per commit harus berupa bilangan riil atau bilangan bulat**
Pesan ini muncul jika argumen dari *seconds-per-commit* bukan bertipe bilangan riil atau bilangan bulat.
- **Path gambar tidak valid**
Pesan ini muncul jika *path* gambar pada argumen *option logo* tidak ditemukan atau *path* gambar bukan merupakan *file* gambar.
- **Panjang commit ID awal harus berada di antara 7-40 karakter**
Pesan ini muncul jika panjang Commit ID pada argumen *option start-commit* kurang dari tujuh karakter atau lebih besar dari empat sepuluh karakter.
- **Panjang commit ID akhir harus berada di antara 7-40 karakter**
Pesan ini muncul jika panjang Commit ID pada argumen *option stop-commit* kurang dari tujuh karakter atau lebih besar dari empat sepuluh karakter.
- **Commit ID awal tidak ditemukan**
Pesan ini muncul jika Commit ID pada argumen *option start-commit* tidak ditemukan.
- **Commit ID akhir tidak ditemukan**
Pesan ini muncul jika Commit ID pada argumen *option stop-commit* tidak ditemukan.

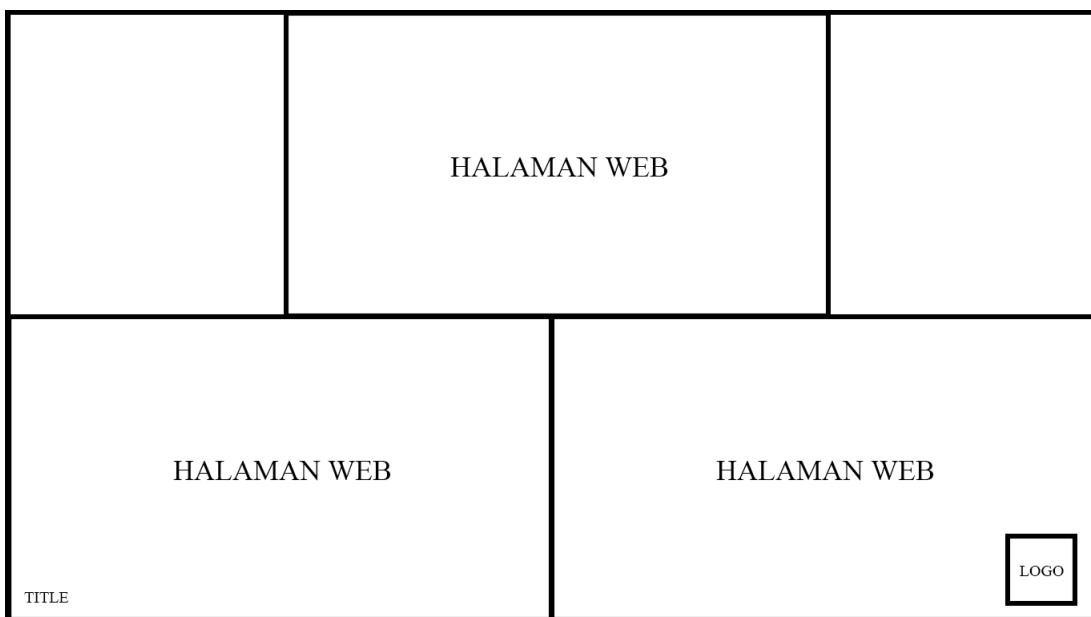
- Commit ID awal dan akhir tidak boleh sama
Pesan ini muncul jika argumen pada *option start-commit* dan *option stop-commit* bernilai sama.
- Commit ID awal dan akhir terbalik
Pesan ini muncul jika nilai argumen pada *option start-commit* dan *option stop-commit* tertukar.
- Branch tidak valid
Pesain ini muncul jika nama branch pada argumen *option branch* tidak valid.



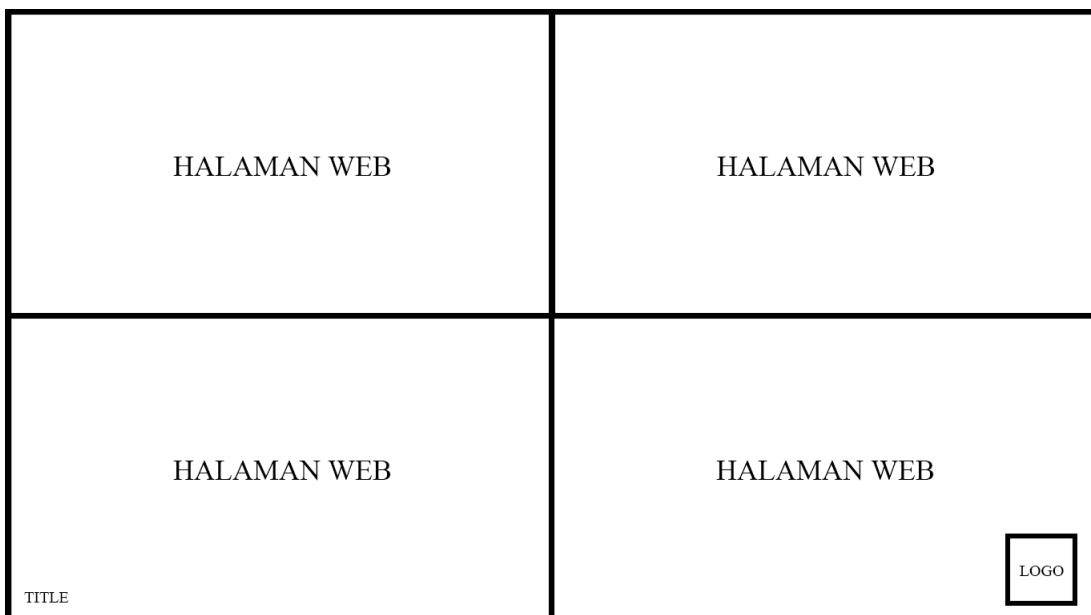
Gambar 4.2: Rancangan tampilan *screenshot* pada *file* hasil animasi jika terdapat satu halaman *web*.



Gambar 4.3: Rancangan tampilan *screenshot* pada *file* hasil animasi jika terdapat dua halaman *web*.



Gambar 4.4: Rancangan tampilan *screenshot* pada file hasil animasi jika terdapat tiga halaman *web*.



Gambar 4.5: Rancangan tampilan *screenshot* pada file hasil animasi jika terdapat empat halaman *web*.

Rancangan tampilan *screenshot* pada file hasil animasi *timelapse* dapat dilihat pada Gambar 4.2 sampai dengan Gambar 4.5. Gambar 4.2 menunjukkan rancangan tampilan *screenshot* pada file hasil animasi jika terdapat satu halaman *web*. Gambar 4.3 menunjukkan rancangan tampilan *screenshot* pada file hasil animasi jika terdapat dua halaman *web*. Gambar 4.4 menunjukkan rancangan tampilan *screenshot* pada file hasil animasi jika terdapat tiga halaman *web*. Gambar 4.5 menunjukkan rancangan tampilan *screenshot* pada file hasil animasi jika terdapat empat halaman *web*.

BAB 5

IMPLEMENTASI DAN PENGUJIAN

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

5.1 Implementasi

5.1.1 Lingkungan Implementasi

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

- Processor: Intel Core i3 4030U
- RAM: 6GB
- Sistem Operasi: Windows 10 pro 64-bit
- Apache HTTP Server versi 2.4.29
- MySQL Server versi 5.5.5
- Netbeans versi 8.1
- Google Chrome versi 73.0.3683.86

5.1.2 Hasil Implementasi

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

1. Status pada Terminal

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

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

```
1 Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on
   port 16446
2 Only local connections are allowed.
3 Feb 24, 2019 3:26:25 PM org.openqa.selenium.remote.ProtocolHandshake
  createSession
```

- 4 | INFO: Detected dialect: OSS
5 | Animasi timelapse berhasil dibuat

2. File GIF Hasil Animasi

Selain menghasilkan status pada terminal, program juga menghasilkan sebuah *file GIF* hasil animasi. Gambar 5.1 menunjukkan *screenshot* halaman *web* setiap *commit* yang terdapat pada *file GIF* hasil animasi dari situs *web* Piktora. Piktora memiliki 58 *commit*, sehingga terdapat 58 *screenshot*.



Gambar 5.1: *Screenshot* halaman *web* setiap *commit* pada situs *web* Piktora.

5.2 Pengujian

5.2.1 Pengujian Fungsional

Pengujian ini dilakukan dengan tujuan untuk mengetahui apakah Command Line Option yang terdapat pada program sudah berjalan dengan baik. *Option* yang terdapat pada program dapat dilihat pada subbab 3.3. Lingkungan pengujian fungsional sama dengan lingkungan implementasi yang terdapat pada subbab 5.1.1. Pengujian dilakukan pada situs *web* Piktora dengan menggunakan berbagai macam tes kasus. *Commit history* situs *web* Piktora dapat dilihat pada subbab 3.4. Hasil pengujian fungsional dapat dilihat pada Tabel 5.1 sampai dengan Tabel 5.3.



Gambar 5.2: Salah satu *screenshot* halaman *web* yang terdapat pada *file* hasil animasi. Terdapat judul di bagian pojok kiri bawah.



Gambar 5.3: Salah satu *screenshot* halaman *web* yang terdapat pada *file* hasil animasi. Terdapat logo di bagian pojok kanan bawah.

Tabel 5.1: Tabel pengujian fungsional

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

Tabel 5.2: Tabel pengujian fungsional

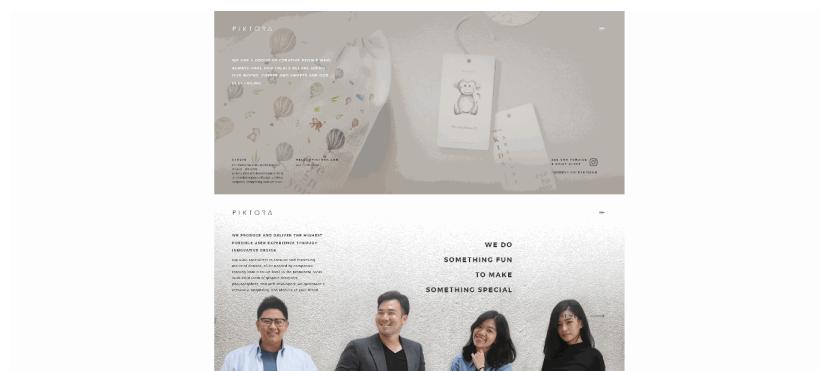
No	Tes Kasus	Output yang diharapkan	Output program
8.	Menguji <i>option -seconds-per-commit</i> dengan nilai 2. Argumen <i>command line</i> yang diberikan ke program: -seconds-per-commit 2	Program berhasil membangkitkan animasi dan menghasilkan <i>file GIF</i> yang memiliki durasi 116 detik	sesuai
9.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

Tabel 5.3: Tabel pengujian fungsional

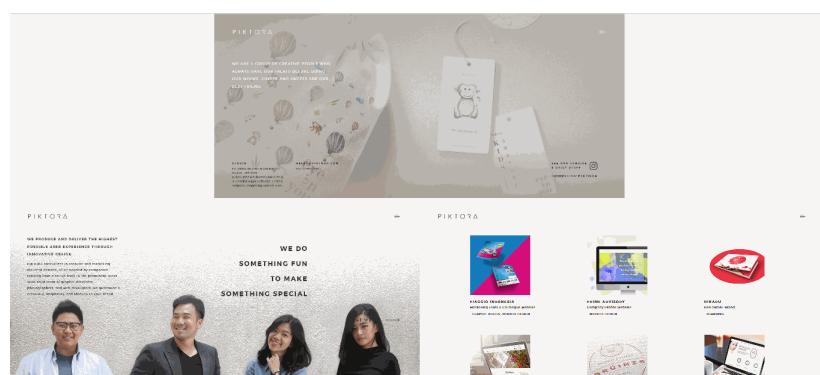
No	Tes Kasus	Output yang diharapkan	Output program
18.	Menguji <i>option -start-commit</i> menggunakan <i>commit ID</i> dengan panjang kurang dari 7 karakter. Argumen <i>command line</i> yang diberikan ke program: -start-commit 9b0a30	Program gagal membuat animasi dan mengeluarkan pesan <i>error</i> "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 <i>command line</i> yang diberikan ke program: -start-commit 9b0a3023ac8c94c67f4c1b39388277768a483dba9	Program gagal membuat animasi dan mengeluarkan pesan <i>error</i> "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 <i>command line</i> yang diberikan ke program: -stop-commit 6a085c1	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 <i>command line</i> yang diberikan ke program: -stop-commit 6a085c	Program gagal membuat animasi dan mengeluarkan pesan <i>error</i> "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 <i>command line</i> yang diberikan ke program: -stop-commit 6a085c1c37949e6308cf06a117302e528388e549	Program gagal membuat animasi dan mengeluarkan pesan <i>error</i> "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 <i>command line</i> yang diberikan ke program: -start-commit 9b0a302 -stop-commit 6a085c1	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 <i>command line</i> yang diberikan ke program: -start-commit 6a085c1 -stop-commit 9b0a302	Program gagal membuat animasi dan mengeluarkan pesan <i>error</i> "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 <i>command line</i> yang diberikan ke program: -start-commit 9b0a302 -stop-commit 9b0a302	Program gagal membuat animasi dan mengeluarkan pesan <i>error</i> "Commit ID awal dan akhir tidak boleh sama"	sesuai



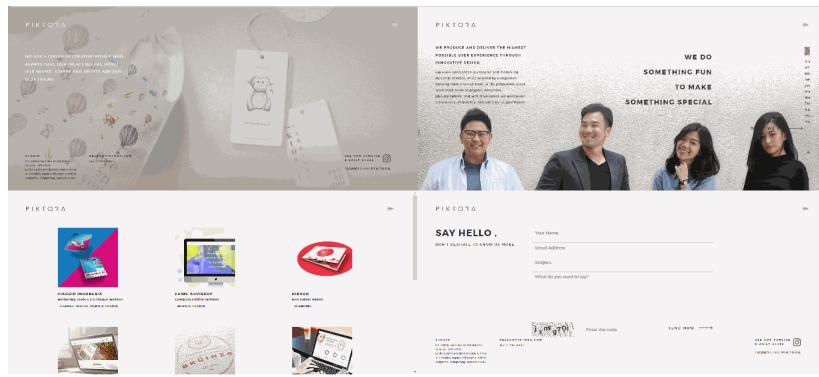
Gambar 5.4: Salah satu *screenshot* halaman *web* pada *file* hasil animasi jika terdapat satu argumen `-capture-url`.



Gambar 5.5: Salah satu *screenshot* halaman *web* hasil animasi jika terdapat dua argumen `-capture-url`.



Gambar 5.6: Salah satu *screenshot* halaman *web* hasil animasi jika terdapat tiga argumen `-capture-url`.



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

5.2.2 Pengujian Eksperimental

Pengujian eksperimental dilakukan dengan menjalankan program dengan berbagai WebDriver dan situs *web*. WebDriver yang digunakan untuk pengujian ini yaitu FirefoxDriver, EdgeDriver, OperaDriver, dan InternetExplorerDriver. Situs *web* yang digunakan untuk menguji progam yaitu Bootstrap¹, Netflix Open Source Software Center², IBM Open Source³, React⁴, dan Yelp Open Source⁵.

Seperti yang sudah dijelaskan pada subbab 2.4, FirefoxDriver, EdgeDriver, OperaDriver, dan InternetExplorerDriver merupakan implementasi dari WebDriver yang mengontrol *browser* yang mengontrol *browser* tertentu. FirefoxDriver mengontrol Firefox *browser*, OperaDriver mengontrol Opera *browser*, dst. Tujuan dari digunakannya berbagai macam WebDriver ini adalah untuk menguji apakah program dapat membangkitkan animasi *timelapse* pada berbagai macam *browser*.

Pengujian menggunakan berbagai situs *web* dilakukan dengan tujuan menguji program dengan berbagai situs *web* yang berbeda dan memiliki jumlah *commit* yang berbeda. Situs *web* Bootstrap, Netflix Open Source Software Center, IBM Open Source, React, dan Yelp Open Source dipilih karena bersifat Open Source, repositorinya tersimpan pada GitHub, dan dihosting menggunakan GitHub Pages. GitHub Pages⁶ merupakan sebuah layanan *hosting* situs *web* untuk melakukan *hosting* halaman *web* secara langsung dari repositori GitHub.

Berikut adalah rincian dari pengujian eksperimental:

1. Pengujian Situs *Web* Piktora dengan FirefoxDriver

Pengujian pada situs *web* Piktora dilakukan menggunakan FirefoxDriver. Pada saat melakukan pengujian, kode program pada kelas BrowserController baris ke-42 diubah (lihat Lampiran A). *Object* bertipe WebDriver dinisialisasi menggunakan *object* bertipe FirefoxDriver. Versi Firefox *browser* yang digunakan untuk pengujian adalah 66.0.2. Gambar 5.8, menunjukkan tampilan Firefox *browser* saat dikontrol oleh FirefoxDriver. Berikut ini adalah *option* yang digunakan untuk menguji program:

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

¹<https://getbootstrap.com/>

²<https://netflix.github.io/>

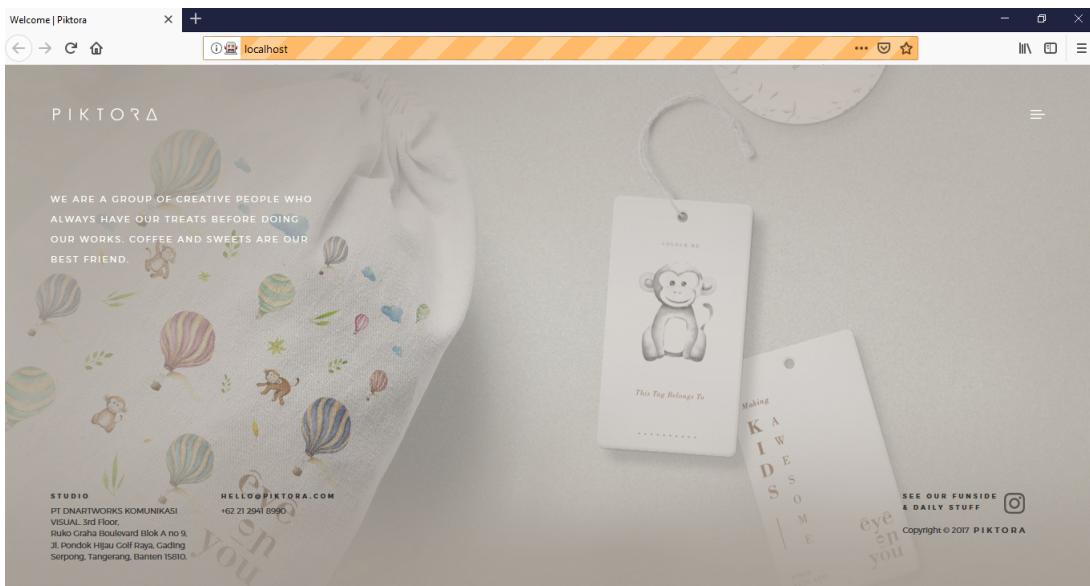
³<https://ibm.github.io/>

⁴<https://reactjs.org/>

⁵<https://yelp.github.io/>

⁶<https://pages.github.com/>

Program berhasil membangkitkan animasi *timelapse* pada situs *web* Piktora menggunakan FirefoxDriver.



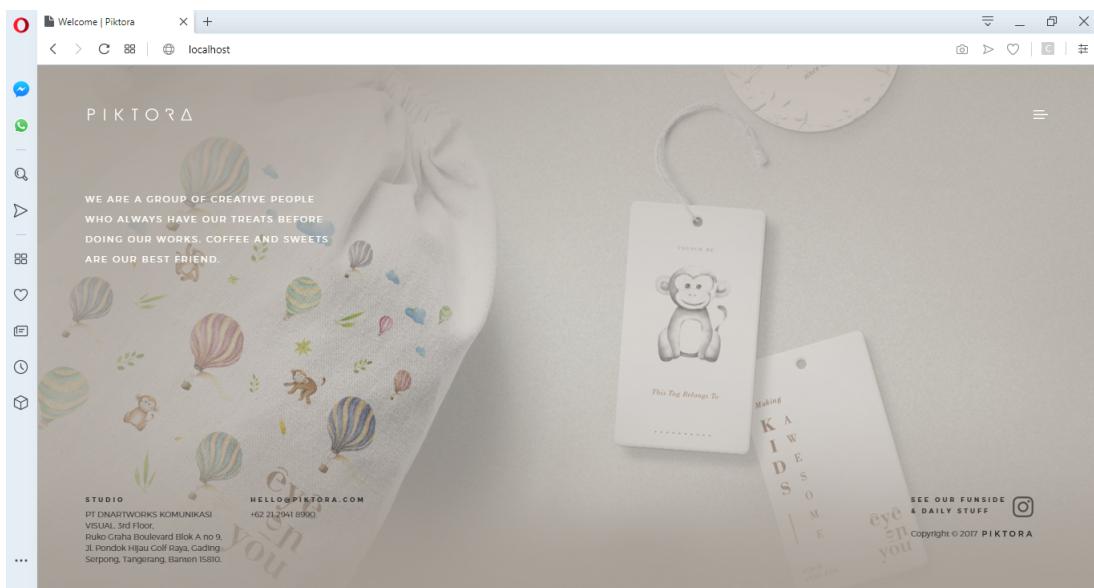
Gambar 5.8: Tampilan pada Firefox *browser* saat dikontrol oleh FirefoxDriver.

2. Pengujian Situs *Web* Piktora dengan OperaDriver

Pengujian pada situs *web* Piktora dilakukan menggunakan OperaDriver. Pada saat melakukan pengujian, kode program pada kelas BrowserController baris ke-42 diubah (lihat Lampiran A). *Object* bertipe WebDriver dinisialisasi menggunakan *object* bertipe OperaDriver. Versi Opera *browser* yang digunakan untuk pengujian adalah 60.0.3255.27 (portable version). Gambar 5.9 menunjukkan tampilan pada Opera *browser* saat dikontrol oleh OperaDriver. Berikut ini adalah *option* yang digunakan untuk menguji program:

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

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



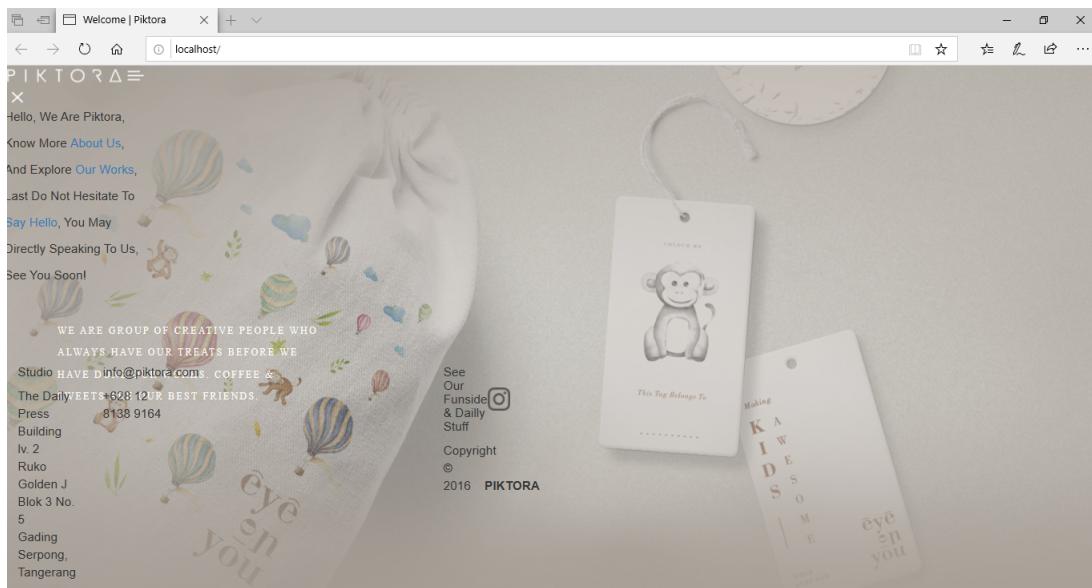
Gambar 5.9: Tampilan pada Opera *browser* saat dikontrol oleh OperaDriver.

3. Pengujian situs *web* Piktora dengan EdgeDriver

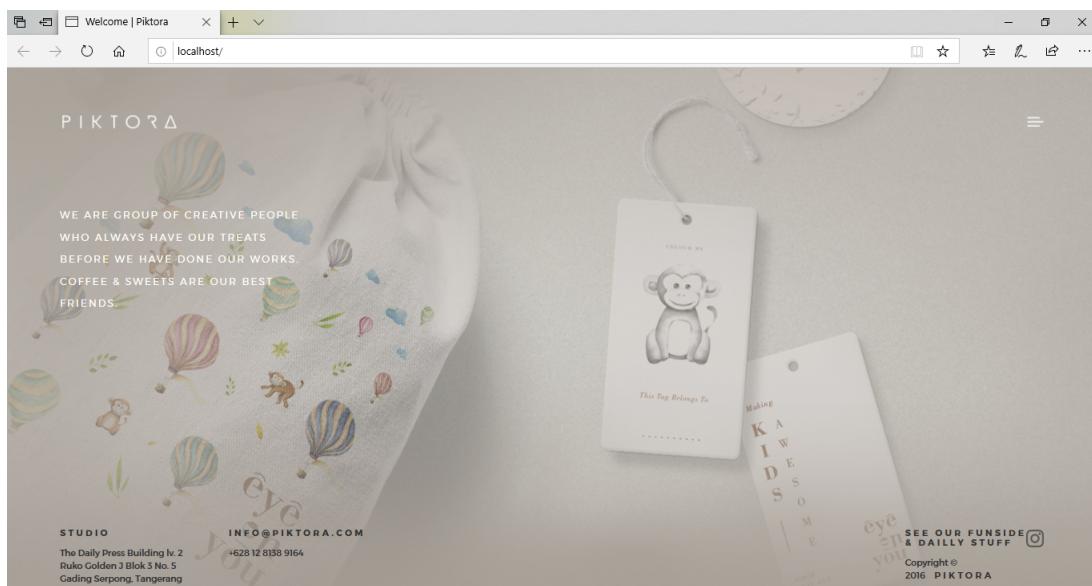
Pengujian pada situs *web* Piktora dilakukan menggunakan EdgeDriver. Pada saat melakukan pengujian, kode program pada kelas BrowserController baris ke-42 diubah (lihat Lampiran A). *Object* bertipe WebDriver dinisialisasi menggunakan *object* bertipe EdgeDriver. Versi Microsoft Edge *browser* yang digunakan untuk pengujian adalah 44.17763.1.0. Berikut ini adalah *option* yang digunakan untuk menguji program:

- -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 situs *web* 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 Situs *Web* Piktora dengan InternetExplorer

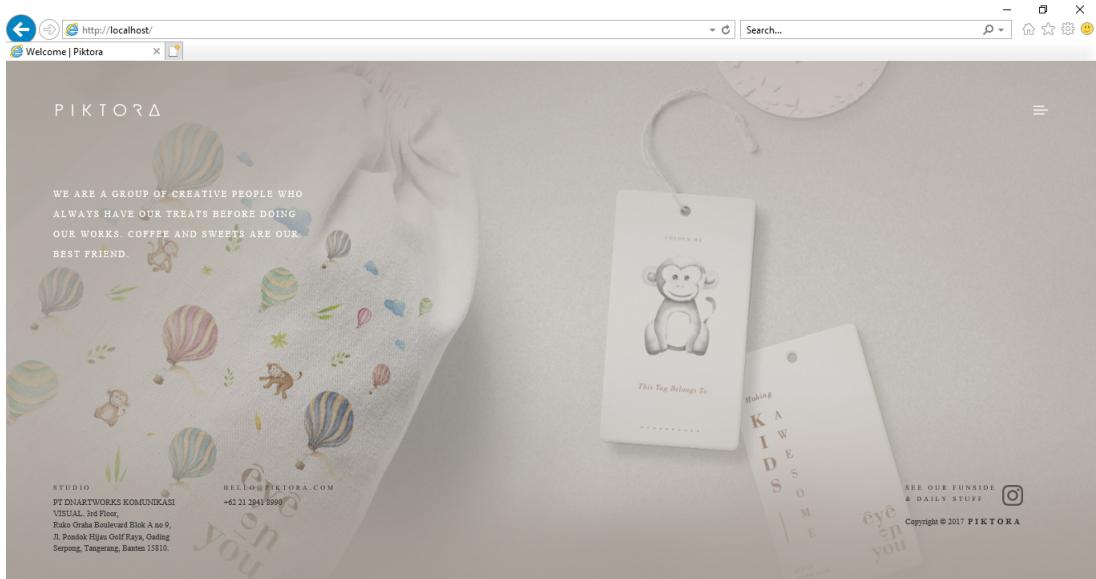
Pengujian pada situs *web* Piktora dilakukan menggunakan InternetExplorerDriver. Pada saat melakukan pengujian, kode program pada kelas BrowserController baris ke-42 diubah (lihat Lampiran A). *Object* bertipe WebDriver dinisialisasi menggunakan *object* bertipe InternetExplorerDriver. Versi Internet 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:

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

Program berhasil membangkitkan animasi *timelapse* pada situs *web* Piktora menggunakan 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

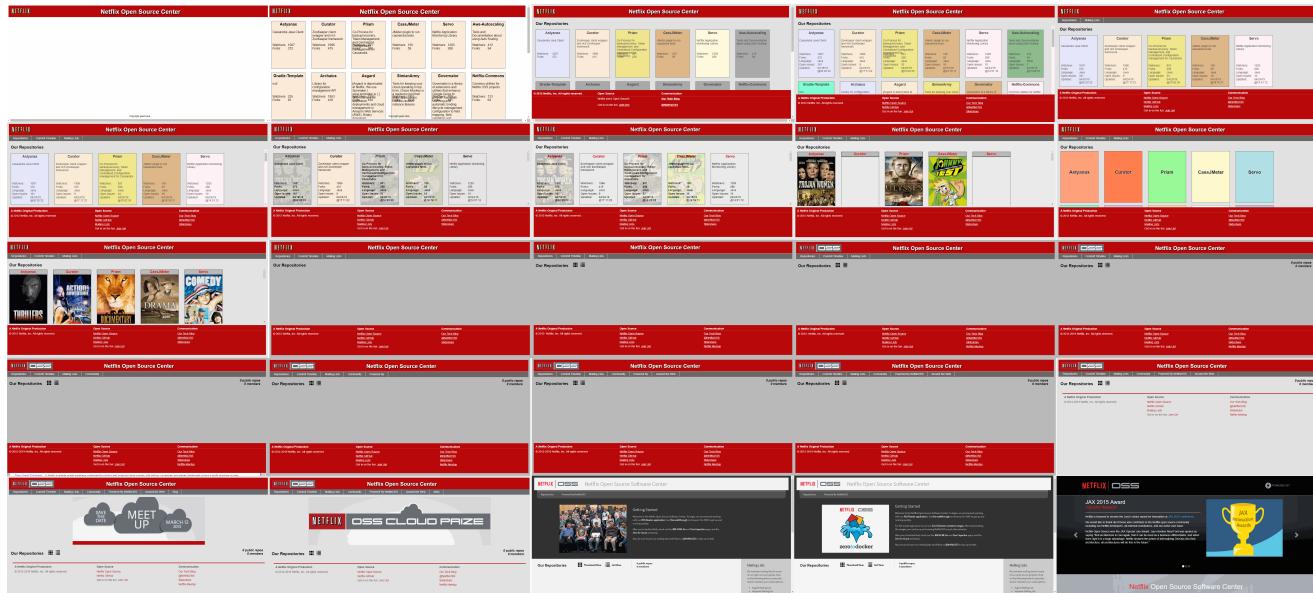
Netflix Open Source Software Center merupakan proyek Open Source yang dimiliki oleh Netflix. Repotori situs *web* ini disimpan pada GitHub⁷. Repotori ini memiliki 393 *commit*. Lingkungan pengujian eksperimental ini sama dengan lingkungan implementasi yang terdapat pada subbab 5.1.1. Berikut ini adalah *option* yang digunakan untuk menguji program:

- `-path C:/xampp/htdocs/netflix.github.com/.git`
- `-capture-url http://localhost`

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

- `-seconds-per-commit 0.1`

Program berhasil membangkitkan animasi *timelapse* dari situs *web* Netflix Open Source Software Center. Tidak ditemukan masalah saat melakukan pengujian. Hasil dari pengujian berupa *file* hasil animasi bertipe GIF dengan ukuran 17.5 MB dan berdurasi 39 detik. Sebagian dari hasil animasi dapat dilihat pada Gambar 5.14, sedangkan *file* aslinya dapat dilihat pada *link* berikut⁸.



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

6. Pengujian Situs *web* Bootstrap

Bootstrap adalah kelas *open source* untuk membangun *website* yang dipakai bersama dengan HTML, CSS, dan JavaScript. Repozitori situs *web* ini disimpan pada GitHub⁹. Pengujian ini dilakukan pada *branch* gh-pages dimana pada *branch* tersebut terdapat 8547 *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:

- `-path C:/xampp/htdocs/bootstrap/.git`
- `-capture-url http://localhost`
- `-seconds-per-commit 0.05`
- `-branch gh-pages`

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

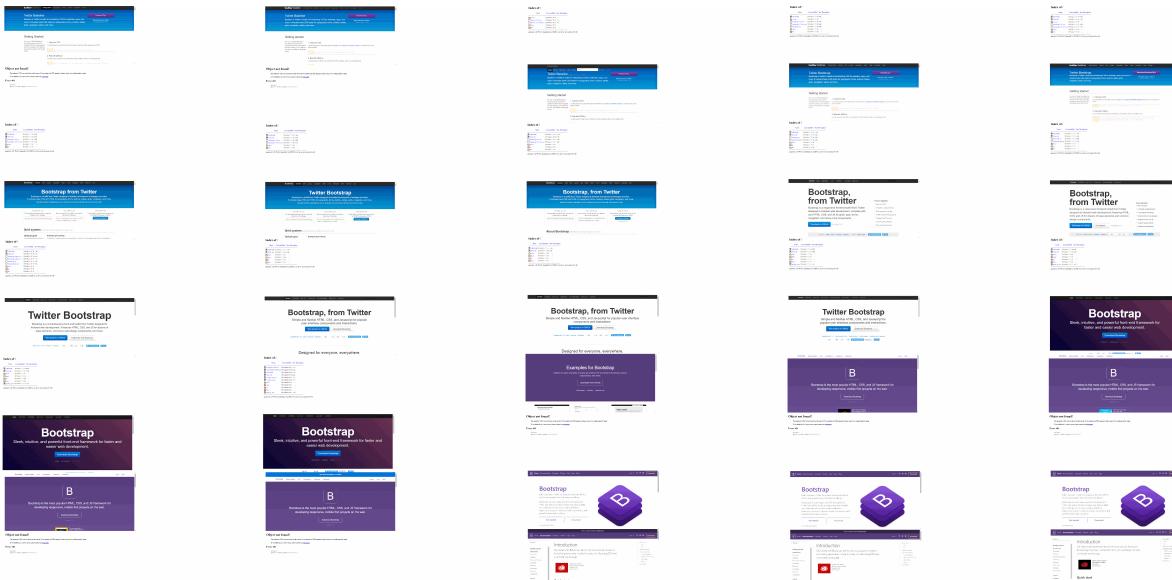
Pada beberapa *commit*, *file* "index.html" tidak terdapat pada *root directory*. Pada beberapa *commit*, *file* "index.html" terletak pada direktori "docs". Karena tidak terdapat *file* "index.html"

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

⁹<https://github.com/twbs/bootstrap>

halaman *web* menjadi tidak muncul, yang muncul adalah struktur direktori dari repositori Bootstrap.

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

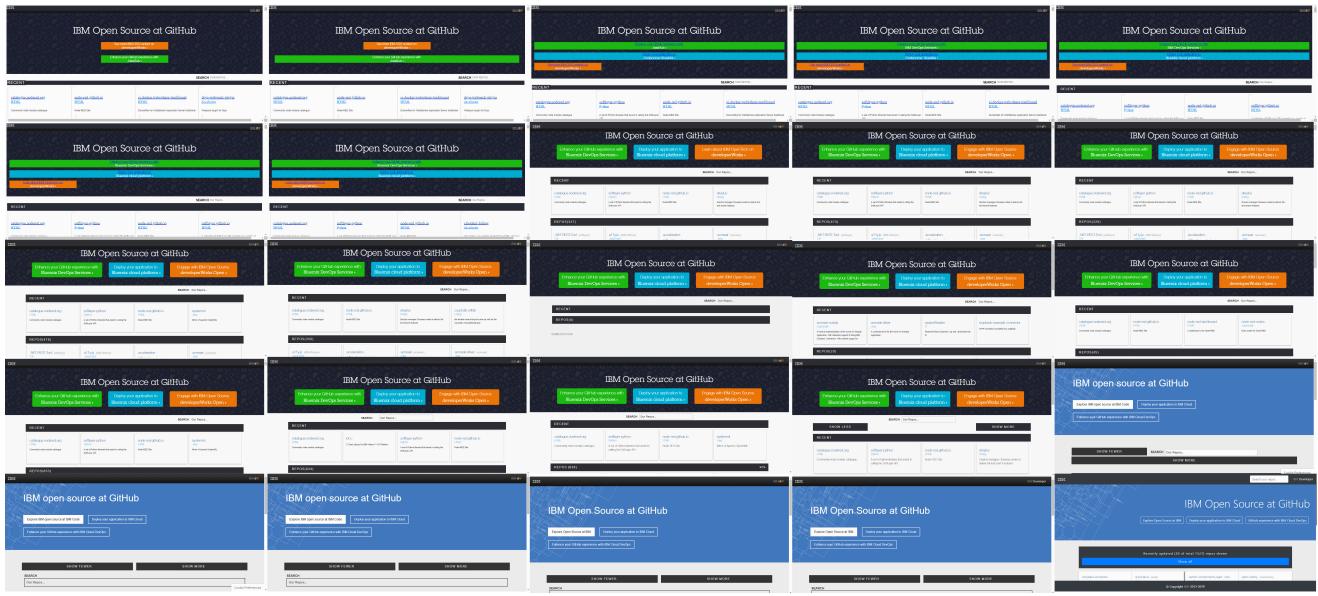
IBM Open Source merupakan proyek Open Source yang dimiliki oleh IBM. Repositori situs *web* ini disimpan pada GitHub¹⁰. Repositori ini memiliki 263 *commit*. Lingkungan pengujian eksperimental ini sama dengan lingkungan implementasi yang terdapat pada subbab 5.1.1. Berikut ini adalah *option* yang digunakan untuk menguji program:

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

Program berhasil membangkitkan animasi *timelapse* dari situs *web* IBM Open Source. Tidak ditemukan masalah saat melakukan pengujian. Hasil dari pengujian berupa *file* hasil animasi bertipe GIF dengan ukuran 15.2 MB dan berdurasi 26 detik. Sebagian dari hasil animasi dapat dilihat pada Gambar 5.16, sedangkan *file* aslinya dapat dilihat pada *link* berikut¹¹.

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

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



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

8. Pengujian Situs *web* React

React merupakan *library Java Script* yang digunakan untuk membangun antarmuka. Reposisori situs *web* ini disimpan pada GitHub¹². Reposisori ini memiliki 570 *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:

- -path C:/xampp/htdocs/react/.git
- -capture-url http://localhost/react
- -seconds-per-commit 0.05
- -branch gh-pages

Program berhasil membangkitkan animasi *timelapse* dari situs *web* React. Untuk beberapa *commit* tidak terdapat *file* "index.html". Hasil dari pengujian berupa *file* hasil animasi bertipe GIF dengan ukuran 18.2 MB dan berdurasi 28 detik. Sebagian dari hasil animasi dapat dilihat pada Gambar 5.17, sedangkan *file* aslinya dapat dilihat pada *link* berikut¹³.

¹²<https://github.com/facebook/react>

¹³<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

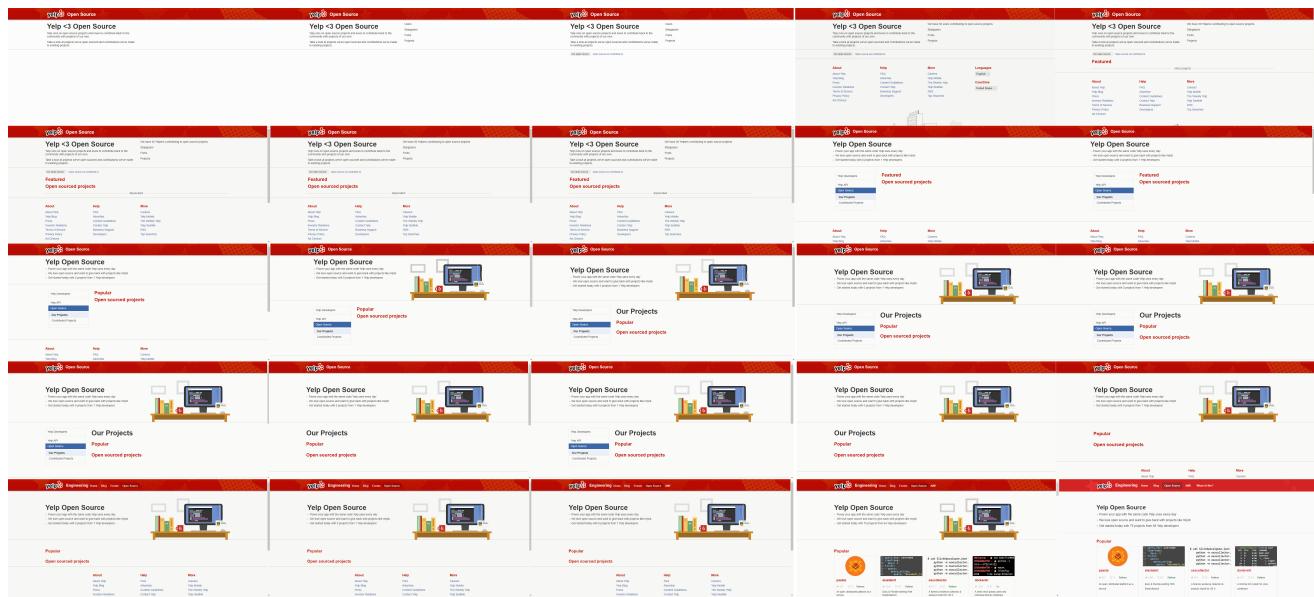
Yelp Open Source merupakan proyek Open Source yang dimiliki oleh Yelp. Reposisori situs *web* ini disimpan pada GitHub¹⁴. Reposisori ini memiliki 99 commit. Lingkungan pengujian eksperimental ini sama dengan lingkungan implementasi yang terdapat pada subbab 5.1.1. Berikut ini adalah *option* yang digunakan untuk menguji program:

- `-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* berikut¹⁵.

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

¹⁵<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 pada 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

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

Hal yang menarik dari pengujian eksperimental adalah letak *file* "index.html" yang tidak konsisten pada repositori situs *web* Bootstrap. Pada beberapa *commit*, *file* "index.html" berada pada *root directory*. Pada beberapa *commit*, *file* "index.html" berada pada direktori "docs". Pada beberapa *commit* tidak terdapat *file* "index.html". Pada beberapa *commit* terakhir, saat membuka *file* "index.html" pada direktori "docs", halaman menjadi dialihkan ke situs *web* Bootstrap¹⁶.

Berdasarkan hasil pengujian eksperimental, didapatkan kesimpulan sebagai berikut:

1. *File* GIF hasil animasi berukuran cukup besar. Pada situs *web* Bootstrap, *file* GIF hasil animasi berukuran 160 MB.
2. Untuk repositori situs *web* yang sederhana seperti Netflix Open Source Center, Bootstrap, IBM Open Source, React, dan Yelp Open Source, tidak dibutuhkan *option -before-capture* untuk membangkitkan animasi. Untuk repositori yang memerlukan setup basis data seperti Piktora, dibutuhkan *option -before-capture* untuk membangkitkan animasi.
3. Program dapat membangkitkan animasi dengan baik saat menggunakan OperaDriver, FirefoxDriver, dan InternetExplorerDriver. Saat dilakukan pengujian menggunakan EdgeDriver, tampilan dari halaman *web* menjadi tidak rapih.

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

BAB 6

KESIMPULAN DAN SARAN

6.1 Kesimpulan

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

- 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.
- Aplikasi untuk membangkitkan *timelapse* pada pengembangan proyek perangkat lunak berbasis *web* dapat diimplementasi dengan bantuan *library* JGit, Selenium WebDriver, dan Apache Commons CLI.
- Program sudah berjalan dengan baik dan dapat membangkitkan animasi *timelapse* dari pengembangan proyek perangkat lunak berbasis *web*. Hal ini ditunjukkan dengan keberhasilan program dalam membangkitkan animasi pada beberapa situs *web* saat dilakukan pengujian eksperimental.

6.2 Saran

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

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. Dalam pengembangan berikutnya, sebaiknya menggunakan format *file* yang lebih modern seperti MP4 atau format lainnya.
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.
3. Selama beberapa detik terdakang tidak terdapat perubahan pada *screenshot* yang terdapat pada *file* GIF hasil animasi. Dalam perkembangan berikutnya, sebaiknya ditambahkan fitur untuk menandai perubahan pada *screenshot* (*highlight*) dan fitur untuk menampilkan tanggal pada setiap *screenshot*.

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] Higgins, M. (2016) *Time-Lapse Photography: Art and Techniques*. The Crowood Press Ltd.
- [5] 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].
- [6] Selenium documentation. <https://www.seleniumhq.org/docs/>. [Online; diakses 17-September-2018].
- [7] Generated documentation. <https://seleniumhq.github.io/selenium/docs/api/java/>. [Online; diakses 17-September-2018].
- [8] Commons - home. <https://commons.apache.org/proper/commons-cli/index.html>. [Online; diakses 11-Oktober-2018].
- [9] Apache commons cli 1.3.1 api. <https://commons.apache.org/proper/commons-cli/javadocs/api-release/index.html>. [Online; diakses 11-Oktober-2018].
- [10] 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 seukuran layar, dan menutup browser. Hasil screenshot disimpan dalam bentuk List of File.
15 *
16 * @author Billy Adiwijaya
17 */
18 public class BrowserController {
19
20     private final WebDriver[] drivers;
21     private final List<File> screenshotFiles;
22     private final int numberOfWorkers;
23
24     /**
25      * Constructor yang berfungsi untuk menginisialisasi variabel yang dimiliki
26      * oleh kelas ini.
27      *
28      * @param numberOfWorkers jumlah browser.
29      */
30     public BrowserController(int numberOfWorkers) {
31         this.drivers = new WebDriver[numberOfWorkers];
32         this.screenshotFiles = new ArrayList<>();
33         this.numberOfWorkers = numberOfWorkers;
34     }
35
36     /**
37      * Method ini berfungsi untuk membuka semua browser, kemudian mengatur
38      * ukuran window browser menjadi seukuran layar.
39      */
40     public void open() {
41         for (int i = 0; i < numberOfWorkers; i++) {
42             this.drivers[i] = new ChromeDriver();
43             this.drivers[i].manage().window().maximize();
44         }
45     }
46
47     /**
48      * Method ini berfungsi untuk mengembalikan jumlah browser yang dimiliki
49      * kelas ini.
50      *
51      * @return jumlah browser yang dimiliki oleh kelas ini.
52      */
53     public int getNumberOfWorkers() {
54         return numberOfWorkers;
55     }
56
57     /**
58      * Method ini berfungsi untuk berpindah halaman pada browser tertentu.
59      *
60      * @param browserIndex indeks browser yang akan diubah halamannya.
61      * @param url alamat URL untuk berpindah halaman.
62      */
63     public void changePage(int browserIndex, String url) {
64         this.drivers[browserIndex].get(url);
65     }
66
67     /**
68      * Method ini berfungsi untuk menutup semua browser.
69      */
70     public void quit() {
71         for (WebDriver driver : this.drivers) {
72             driver.quit();
73         }
74     }
75 }
```

```

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

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.
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 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
35             dicapture").build());
36         options.addOption(Option.builder().required().longOpt("path").argName("path").hasArg().desc("path repositori perangkat
37             lunak berbasis web").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 mengambil screenshot").build());
42         options.addOption(Option.builder().longOpt("start-commit").argName("commit id").hasArg().desc("commit id awal untuk
43             membangkitkan animasi").build());
44         options.addOption(Option.builder().longOpt("stop-commit").argName("commit id").hasArg().desc("commit id akhir untuk
45             membangkitkan 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 dicapture maksimal 4");
56         }
57     }
58
59     /**
60     * Method ini berfungsi untuk mengembalikan option yang sudah diparsing.
61     *
62     * @return option yang sudah diparsing berupa objek dengan tipe Properties.
63     */
64     public Properties getParsedOptions() {
65         Properties properties = new Properties();
66         for (Option option : this.commandLine.getOptions()) {
67             if (option.getLongOpt().equals("capture-url")) {
68                 String[] values = option.getValues();
69                 String value = values[0];
70
71                 for (int i = 1; i < values.length; i++) {
72                     value = value + ";" + values[i];
73                 }
74                 properties.setProperty(option.getLongOpt(), value);
75             } else {
76                 properties.setProperty(option.getLongOpt(), option.getValue());
77             }
78         }
79     }
80 }
```

```

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 numberOfWorkers = 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("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(numberOfWorkers);
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      * repositori 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<>();
80         List<File> resultImages = new ArrayList<>();
81         screenshotFiles = browserController.getScreenshotFiles();
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.getType());
93
94             Graphics2D graphics = bufferedImageResult.createGraphics();
95             graphics.setBackground(Color.WHITE);
96             graphics.fillRect(0, 0, bufferedImageResult.getWidth(), bufferedImageResult.getHeight());
97
98             graphics.drawImage(bufferedImage1, bufferedImage1.getWidth() / 4, 0, bufferedImage1.getWidth() / 2,
99                                 bufferedImage1.getHeight() / 2, null);
100            graphics.drawImage(bufferedImage2, bufferedImage2.getWidth() / 4, bufferedImage2.getHeight() / 2,
101                            bufferedImage2.getWidth() / 2, bufferedImage2.getHeight() / 2, null);
102            graphics.dispose();
103            ImageIO.write(bufferedImageResult, "png", resultImages.get(i));
104        }
105        break;
106    case 3:
107        for (int i = 0, j = 0; i < screenshotFiles.size() / browserController.getNumberOfBrowser(); i++, j += 3) {
108            resultImages.add(screenshotFiles.get(i));
109            BufferedImage bufferedImage1, bufferedImage2, bufferedImage3, bufferedImageResult;
110            bufferedImage1 = ImageIO.read(screenshotFiles.get(j));
111            bufferedImage2 = ImageIO.read(screenshotFiles.get(j + 1));
112            bufferedImage3 = ImageIO.read(screenshotFiles.get(j + 2));
113            bufferedImageResult = new BufferedImage(bufferedImage1.getWidth(), bufferedImage1.getHeight(), bufferedImage1.getType());
114
115            Graphics2D graphics = bufferedImageResult.createGraphics();
116            graphics.setBackground(Color.WHITE);
117            graphics.fillRect(0, 0, bufferedImageResult.getWidth(), bufferedImageResult.getHeight());
118
119            graphics.drawImage(bufferedImage1, bufferedImage1.getWidth() / 4, 0, bufferedImage1.getWidth() / 2,
120                                bufferedImage1.getHeight() / 2, null);
121            graphics.drawImage(bufferedImage2, 0, bufferedImage2.getHeight() / 2, bufferedImage2.getWidth() / 2,
122                                bufferedImage2.getHeight() / 2, null);
123            graphics.drawImage(bufferedImage3, bufferedImage3.getWidth() / 2, bufferedImage3.getHeight() / 2,
124                                bufferedImage3.getWidth() / 2, bufferedImage3.getHeight() / 2, null);
125            graphics.dispose();
126            ImageIO.write(bufferedImageResult, "png", resultImages.get(i));
127        }
128        break;
129    case 4:
130        for (int i = 0, j = 0; i < screenshotFiles.size() / browserController.getNumberOfBrowser(); i++, j += 4) {
131            resultImages.add(screenshotFiles.get(i));
132            BufferedImage bufferedImage1, bufferedImage2, bufferedImage3, bufferedImage4, bufferedImageResult;
133            bufferedImage1 = ImageIO.read(screenshotFiles.get(j));
134            bufferedImage2 = ImageIO.read(screenshotFiles.get(j + 1));
135            bufferedImage3 = ImageIO.read(screenshotFiles.get(j + 2));
136            bufferedImage4 = ImageIO.read(screenshotFiles.get(j + 3));
137
138            bufferedImageResult = new BufferedImage(bufferedImage1.getWidth(), bufferedImage1.getHeight(), bufferedImage1.getType());
139
140            Graphics2D graphics = bufferedImageResult.createGraphics();
141            graphics.setBackground(Color.WHITE);
142            graphics.fillRect(0, 0, bufferedImageResult.getWidth(), bufferedImageResult.getHeight());
143
144            graphics.drawImage(bufferedImage1, 0, 0, bufferedImage1.getWidth() / 2, bufferedImage1.getHeight() / 2, null);
145            graphics.drawImage(bufferedImage2, bufferedImage2.getWidth() / 2, 0, bufferedImage2.getWidth() / 2,
146                                bufferedImage2.getHeight() / 2, null);
147            graphics.drawImage(bufferedImage3, 0, bufferedImage3.getHeight() / 2, bufferedImage3.getWidth() / 2,
148                                bufferedImage3.getHeight() / 2, null);
149            graphics.drawImage(bufferedImage4, bufferedImage4.getWidth() / 2, bufferedImage4.getHeight() / 2,
150                                bufferedImage4.getWidth() / 2, bufferedImage4.getHeight() / 2, null);
151            graphics.dispose();
152            ImageIO.write(bufferedImageResult, "png", resultImages.get(i));
153        }
154        break;
155    default:
156        break;
157    }
158
159    for (File resultImage : resultImages) {
160        if (properties.getProperty("title") == null && properties.getProperty("logo") == null) {
161            break;
162        } else {
163            if (properties.getProperty("title") != null) {
164                BufferedImage bufferedImage = ImageIO.read(resultImage);
165                Graphics2D graphic = (Graphics2D) bufferedImage.getGraphics();
166                graphic.setFont(new Font("Times New Roman", Font.BOLD, 18));
167                FontMetrics fontMetrics = graphic.getFontMetrics(graphic.getFont());
168                graphic.setColor(Color.black);
169                graphic.drawString(properties.getProperty("title"), 5, bufferedImage.getHeight() - fontMetrics.getDescent() -
170                                  5);
171                graphic.dispose();
172                ImageIO.write(bufferedImage, "png", resultImage);
173            }
174
175            if (properties.getProperty("logo") != null) {
176                BufferedImage logo = ImageIO.read(new File(properties.getProperty("logo")));
177                BufferedImage bufferedImage = ImageIO.read(resultImage);
178                Graphics2D graphic = (Graphics2D) bufferedImage.getGraphics();
179            }
180        }
181    }
182
183    if (properties.getProperty("title") == null && properties.getProperty("logo") == null) {
184        System.out.println("No title or logo provided");
185    }
186
187    System.out.println("Screenshot processing completed");
188
189    return resultImages;
190}

```

```

170         graphic.drawImage(logo, bufferedImage.getWidth() - logo.getWidth() - 5, bufferedImage.getHeight() - logo.
171                         getHeight() - 5, null);
172         graphic.dispose();
173     }
174 }
175 String fileName = String.format("%s.gif", new SimpleDateFormat("yyyy-MM-dd.HH.mm.ss").format(new Date()));
176 try (ImageOutputStream output = new FileImageOutputStream(new File(fileName))) {
177     int frameDelay = 1000;
178     if (properties.getProperty("seconds-per-commit") != null) {
179         frameDelay = (int) (Double.parseDouble(properties.getProperty("seconds-per-commit")) * 1000);
180         if (frameDelay == 0) {
181             frameDelay = 1000;
182         }
183     }
184     GifSequenceWriter writer = new GifSequenceWriter(output, ImageIO.read(resultImages.get(0)).getType(), frameDelay,
185                                                     false);
186     for (File resultImage : resultImages) {
187         BufferedImage bufferedImage = ImageIO.read(resultImage);
188         writer.writeToSequence(bufferedImage);
189     }
190     writer.close();
191 }
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 dengan repositori 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 commit history repositori perangkat lunak berbasis
30      * web pada branch tertentu. Dimana branch tersebut diambil dari parameter
31      * constructor.
32      *
33      * @param path merupakan path dari repositori perangkat lunak berbasis web.
34      * @param branch nama branch yang digunakan untuk membangkitkan animasi.
35      * @throws IOException jika path repositori 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 repositori 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 /**

```

```
69  * Berfungsi untuk melakukan checkout ke commit tertentu.  
70  *  
71  * @param commitIndex indeks dari variabel commitIDs.  
72  * @throws GitAPIException jika terjadi masalah saat melakukan operasi Git  
73  * Checkout.  
74  */  
75  public void checkoutCommit(int commitIndex) throws GitAPIException {  
76      this.git.checkout().setName(this.commitIDs.get(commitIndex)).call();  
77  }  
78  
79  /**  
80  * Berfungsi untuk melakukan checkout ke commit terakhir yang ada di branch master.  
81  *  
82  * @throws GitAPIException jika terjadi masalah saat melakukan operasi Git  
83  * Checkout.  
84  */  
85  public void checkoutMaster() throws GitAPIException {  
86      this.git.checkout().setName("master").call();  
87  }  
88  
89  /**  
90  * Berfungsi untuk melakukan operasi Git Reset. Operasi ini menghapus  
91  * perubahan pada working directory dan staging area di commit tertentu.  
92  *  
93  * @throws GitAPIException jika terjadi masalah saat melakukan operasi Git  
94  * Reset.  
95  */  
96  public void hardReset() throws GitAPIException {  
97      this.git.reset().setMode(ResetType.HARD).call();  
98  }  
99  
100 /**  
101 * Berfungsi untuk mendapatkan jumlah commit.  
102 *  
103 * @return jumlah commit.  
104 */  
105 public int getNumberOfCommit() {  
106     return this.commitIDs.size();  
107 }  
108  
109 /**  
110 * Berfungsi untuk mendapatkan index dari variabel commitID.  
111 *  
112 * @param commitID merupakan commit ID yang dicari indeksnya.  
113 * @return indeks dari variabel commitIDs.  
114 */  
115 public int getCommitIndex(String commitID) {  
116     int result = -1;  
117     for (int i = 0; i < this.commitIDs.size(); i++) {  
118         if (commitID.equals(this.commitIDs.get(i).substring(0, commitID.length()))) {  
119             result = i;  
120             break;  
121         }  
122     }  
123     return result;  
124 }  
125 }
```


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';
```

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 </style>
99 </head>
100 <body>
101   <?php $this->load->view('templates/navbar'); ?>
102   <div class="container-fluid container-all">
103     <div class="row">
104       <div class="col-xs-12 col-sm-6 col-sm-push-6 col-md-7 col-md-push-5 container-content-img">
105         <?php foreach($photos as $photo): ?>
106           ">
107         <?php endforeach; ?>
```

```

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-->
```