

SISTEM PENCEGAH PLAGIARISME

DALAM TUGAS MAHASISWA DENGAN

SNAPSHOTTING

DAN

USER ACTIVITY LOGGING



Azzam Syawqi Aziz

15515020711132



Agenda

- **Pendahuluan**
 - Latar Belakang & Rumusan Masalah & Tujuan
- **Landasan Kepustakaan**
 - Kajian Pustaka & Landasan Teori
- **Metodologi Penelitian**
- **Pembahasan Hasil**
 - Rekayasa Kebutuhan & Perancangan & Implementasi & Pengujian
- **Kesimpulan & Saran**



Motivasi Plagiarisme

Sumber daya internet dan kemudahan informasi.

(Born, 2003)

Bekerja paruh waktu dan dilema sosial.

(Park, 2004)





Akibat Plagiarisme

Tidak hanya mahasiswa, peneliti dan dosen dari universitas-universitas ternama.

(Agustina & Raharjo, 2017)

Karier tokoh-tokoh besar hancur.

Kiss (2013)



Kelemahan Plagiarism Detection

- Paradigma **polisi-kriminal** (Howard, 2002)
- Pola pikir **nilai oriented** (Dweck & Leggett, 1988)
- Tidak dapat melacak **bantuan eksternal** (Hellas, Leinonen, & Ihantola, 2017)



Kelemahan Plagiarism Detection (I)

- ◉ **Harga** tidak terjangkau (Agustina & Raharjo, 2017; Modiba, Pieterse, & Haskins, 2016)
- ◉ **Pelanggaran hak intelektual** dan hak milik (Park, 2004)

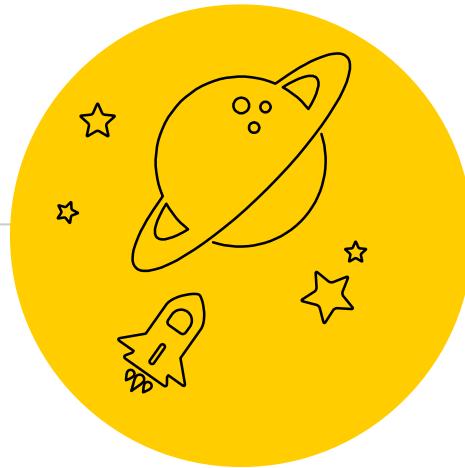


Urgensi untuk Plagiarism Prevention

PDS (Plagiarism Detection Software) tidak dapat mendeteksi semua kasus plagiarisme. (Martins, et al., 2014)

Kemampuan plagiarism prevention tidak dapat dilakukan oleh PDS. (Hellas, Leinonen, & Ihantola, 2017)

Plagiarism prevention sangat diutamakan.
(Garner, et al., 2012)



Plagiarism Prevention





State-of-art Plagiarism Prevention

Plagiarism in Take-home Exams: Helpseeking, Collaboration, and Systematic Cheating.

Hellas, Leinonen, & Ihantola, 2017

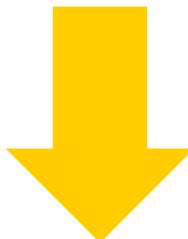


Kelemahan & Solusi

Mengabaikan proses
pengerjaan



Membeli tugas
(Leung & Cheng, 2017)



Snapshottting



Kelemahan & Solusi (I)

Analisis hanya **di akhir**

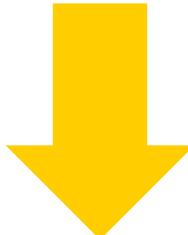


Kecurangan **selama** mengerjakan
(Hellas, Leinonen, & Ihantola, 2017)

Tidak memiliki *log*



Tidak dapat menilai **usaha**
(Hellas, Leinonen, & Ihantola, 2017)



User Activity Logging

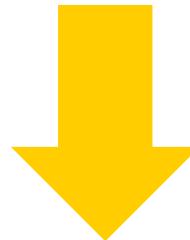


Kelemahan & Solusi (II)

Terikat suatu *platform*



Bentuk tugas & *software* **terbatas**



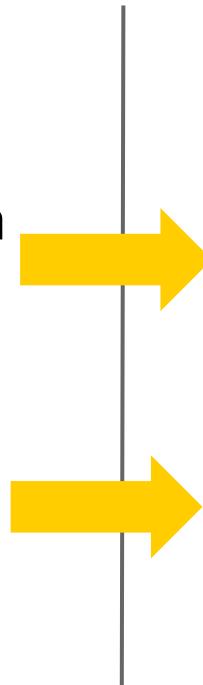
Platform agnostic software



Rumusan Masalah & Tujuan

Rumusan Masalah

- Hasil analisis kebutuhan sistem
- Hasil perancangan & implementasi sistem



Tujuan

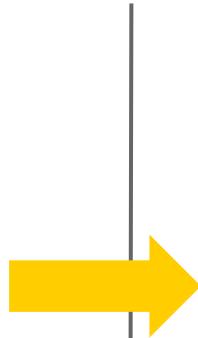
- Menganalisis kebutuhan yang diperlukan sistem
- Merancang & mengimplementasikan sistem sesuai analisis kebutuhan



Rumusan Masalah & Tujuan (I)

Rumusan Masalah

- Hasil pengujian sistem



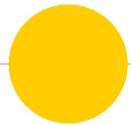
Tujuan

- Memastikan sistem sesuai dengan hasil rancangan dan kebutuhan sebelumnya.



Plagiarism in Take-home Exams: Helpseeking, Collaboration, and Systematic Cheating.

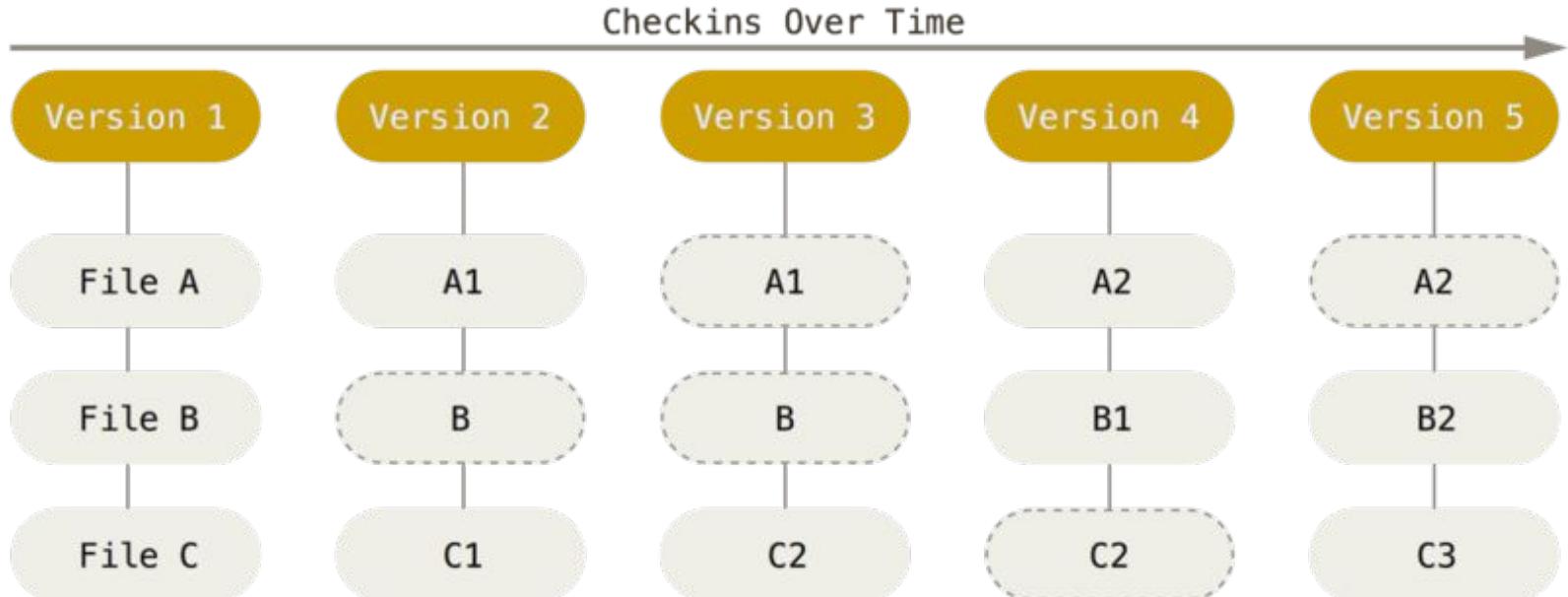
Hellas, Leinonen, & Ihantola, 2017



Landasan Teori

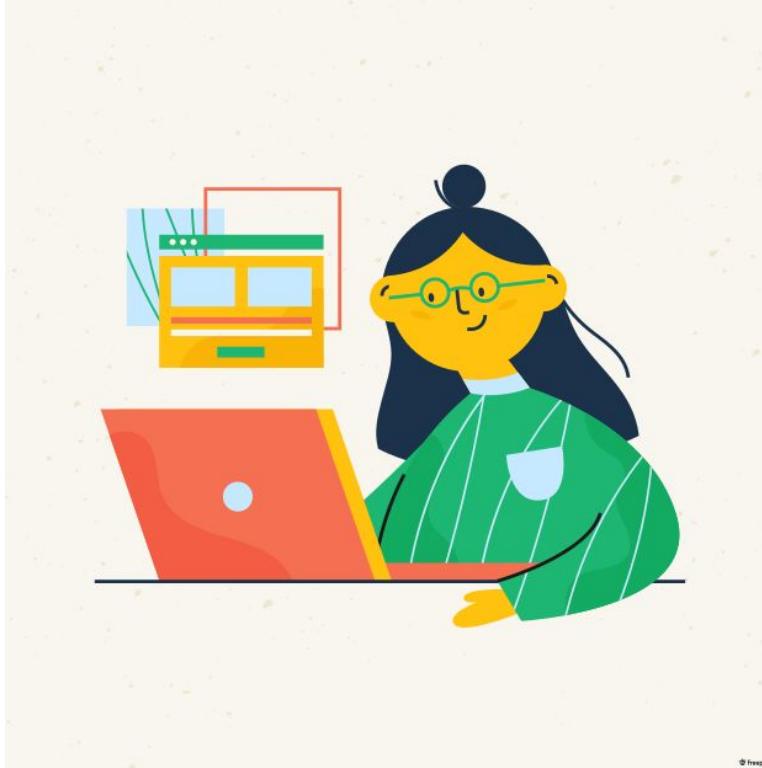


Snapshotting





User Activity Logging

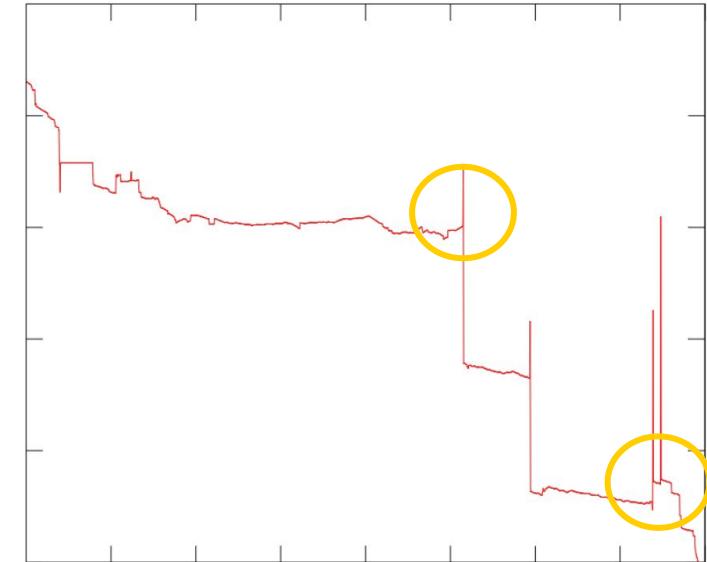


(Macbeth, et al., 2007)



Algoritme Edit-distance

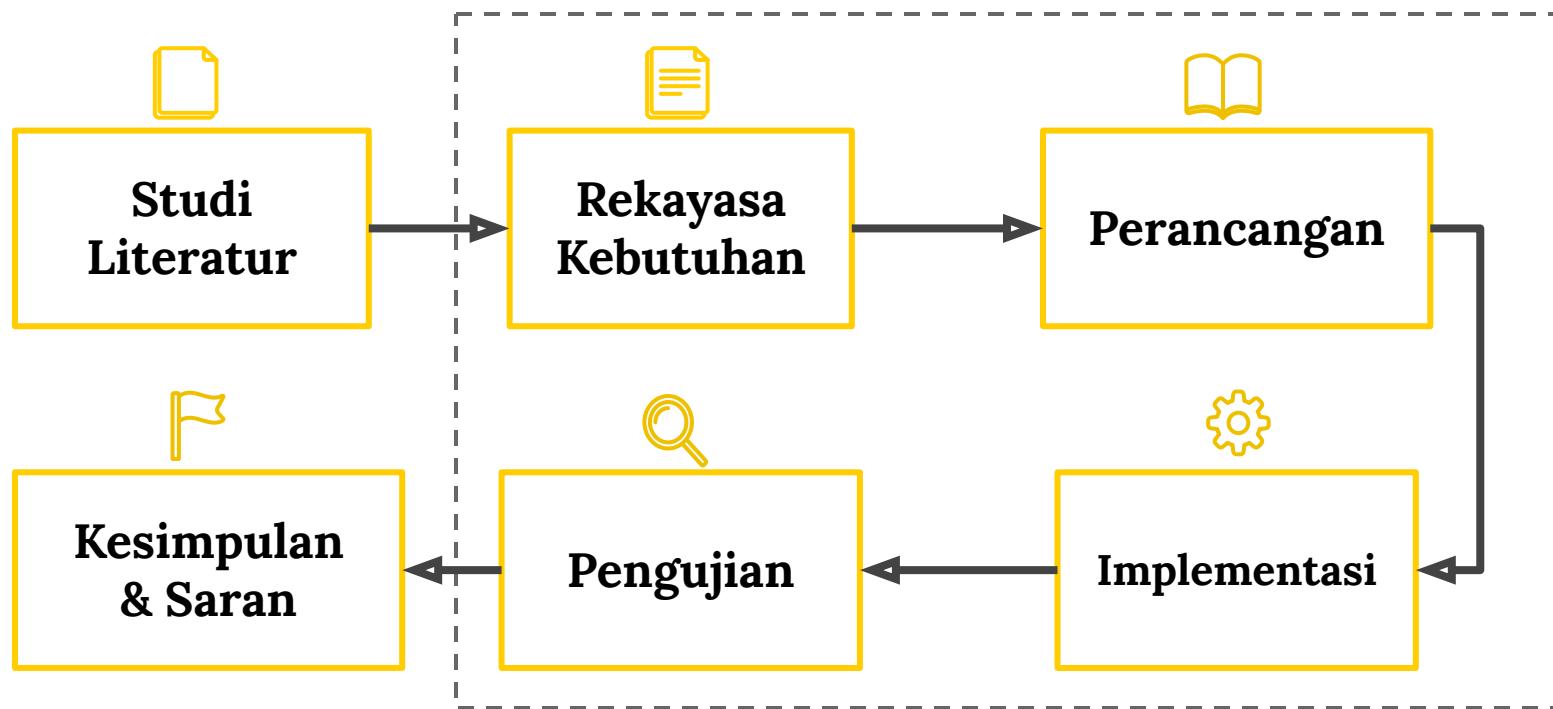
- 1) **tekkom** → fekkom (“t” → “f”)
- 2) fekkom → fíkkom (“e” → “í”)
- 3) fíkkom → **fílkom** (“k” → “l”)





Metodologi Penelitian

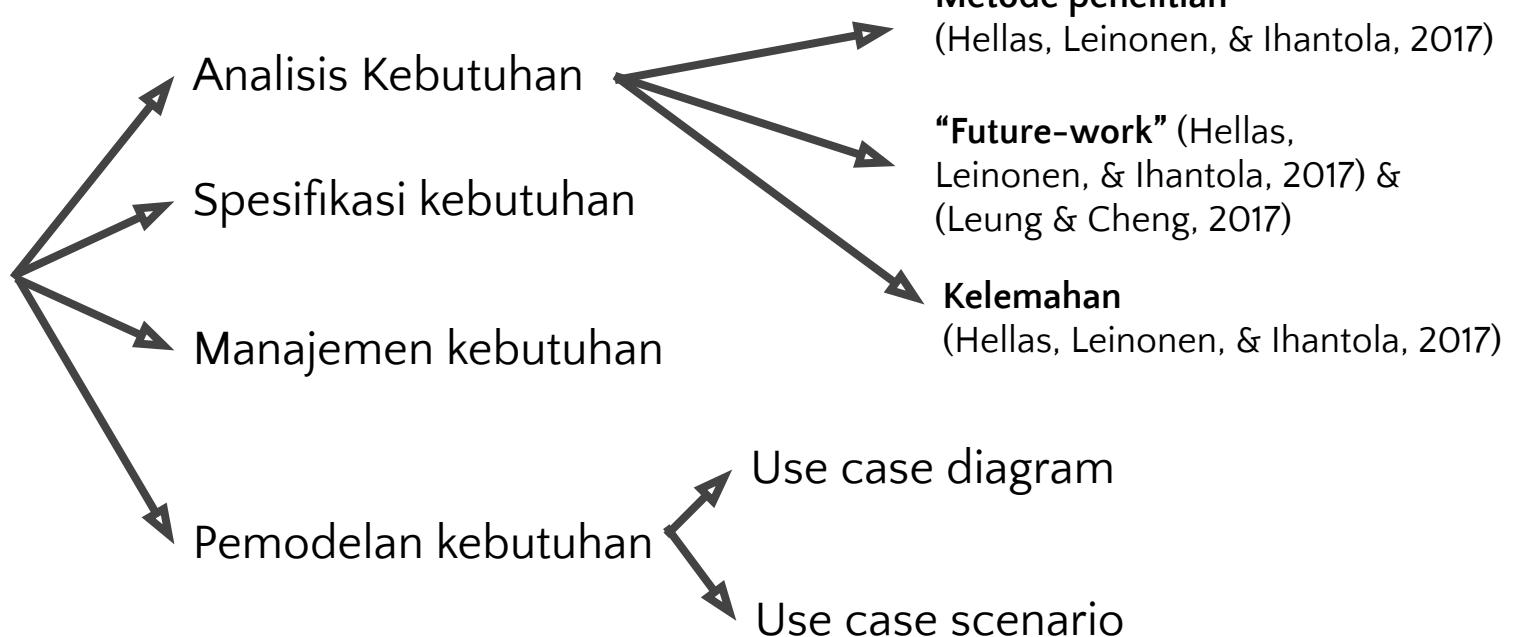
Waterfall Model





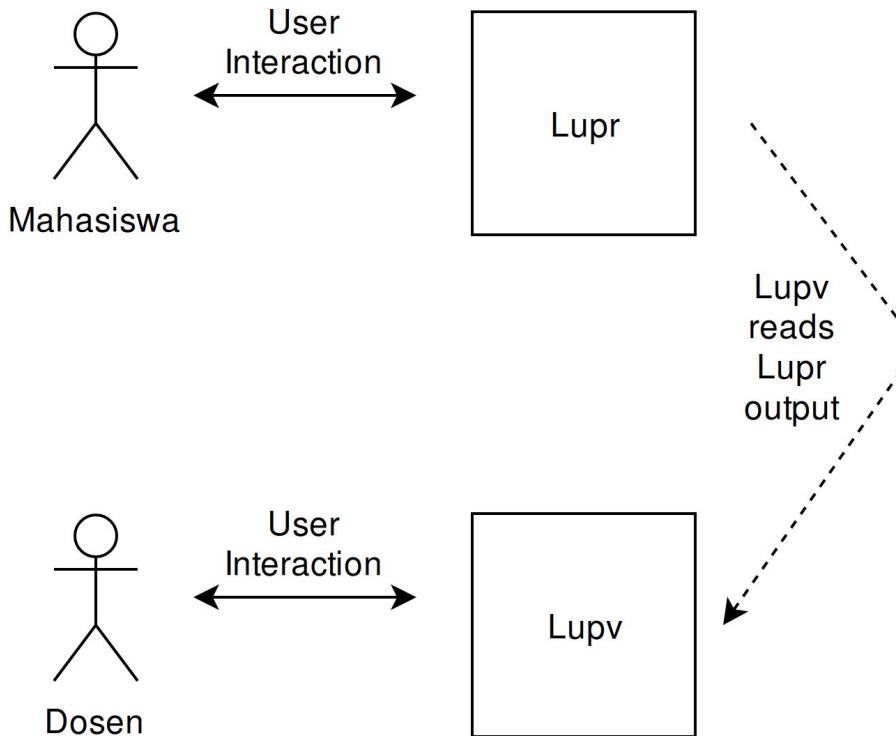
Rekayasa Kebutuhan

Rekayasa Kebutuhan



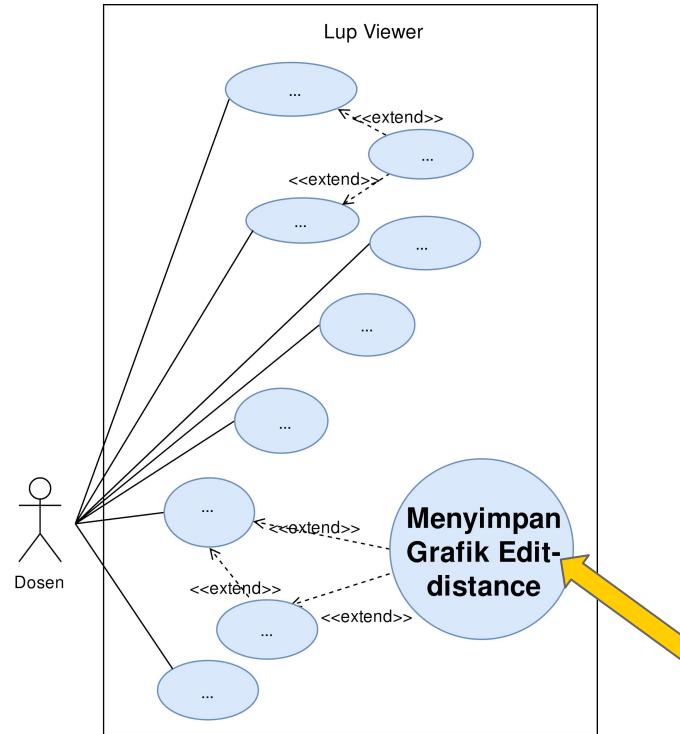


Deskripsi Sistem





Hasil Rekayasa Kebutuhan

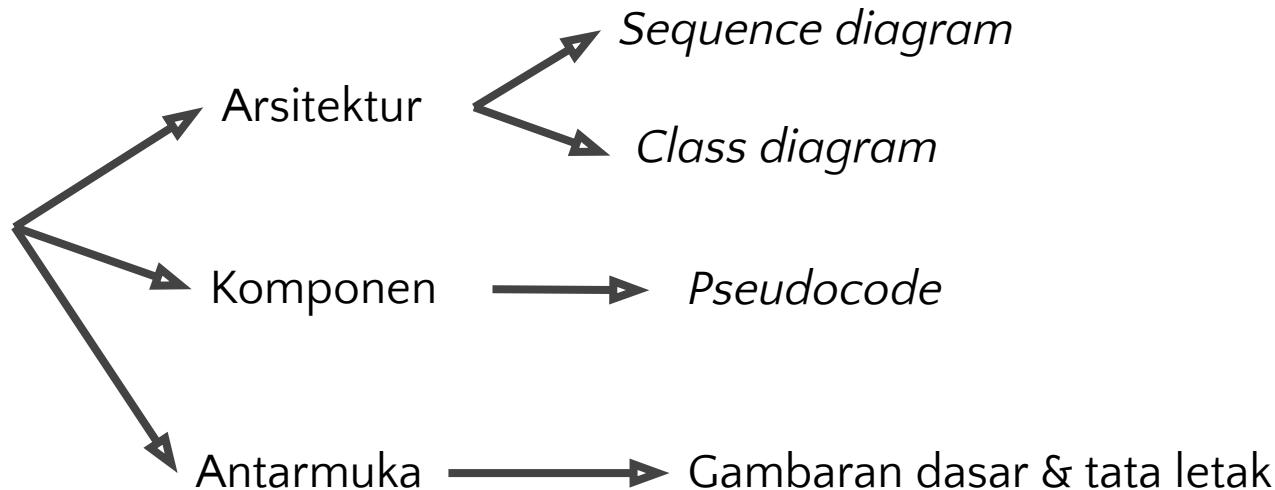




Perancangan

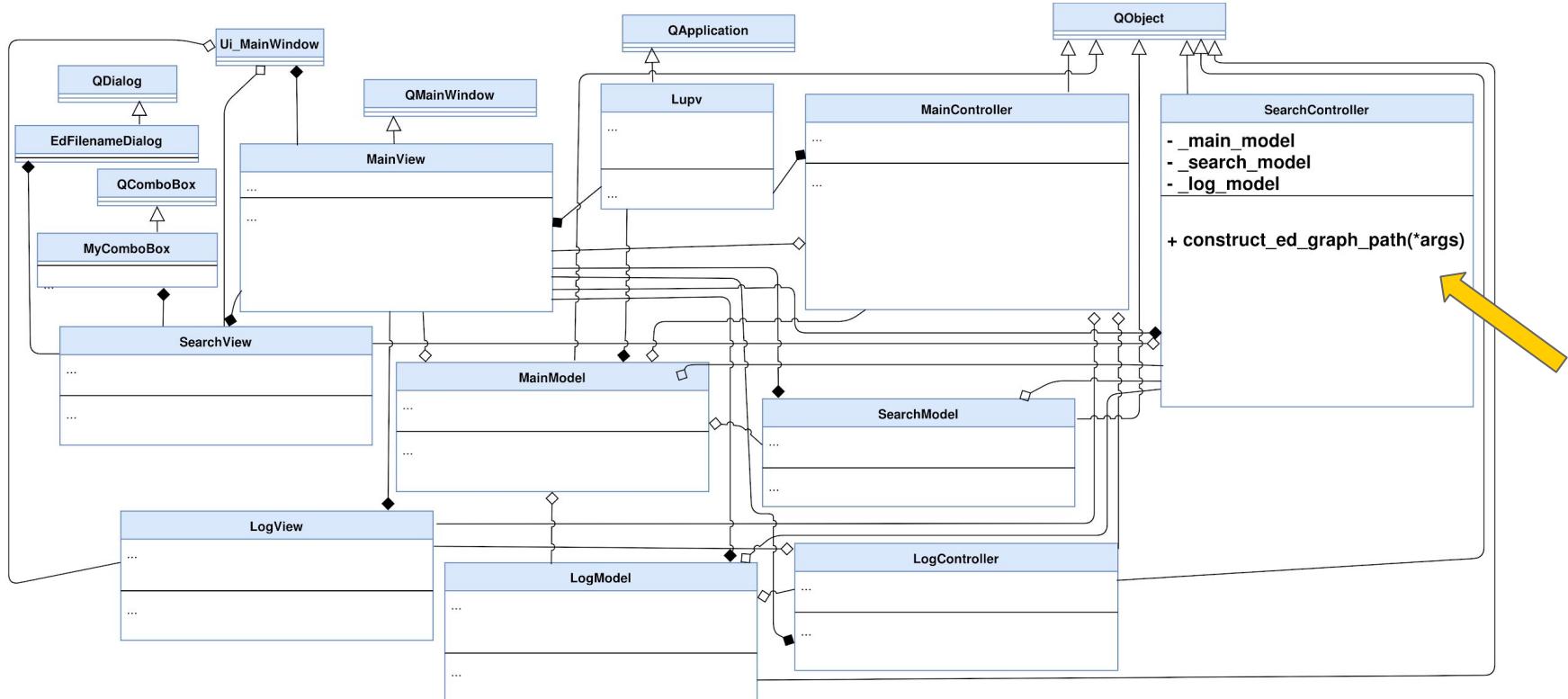


Perancangan

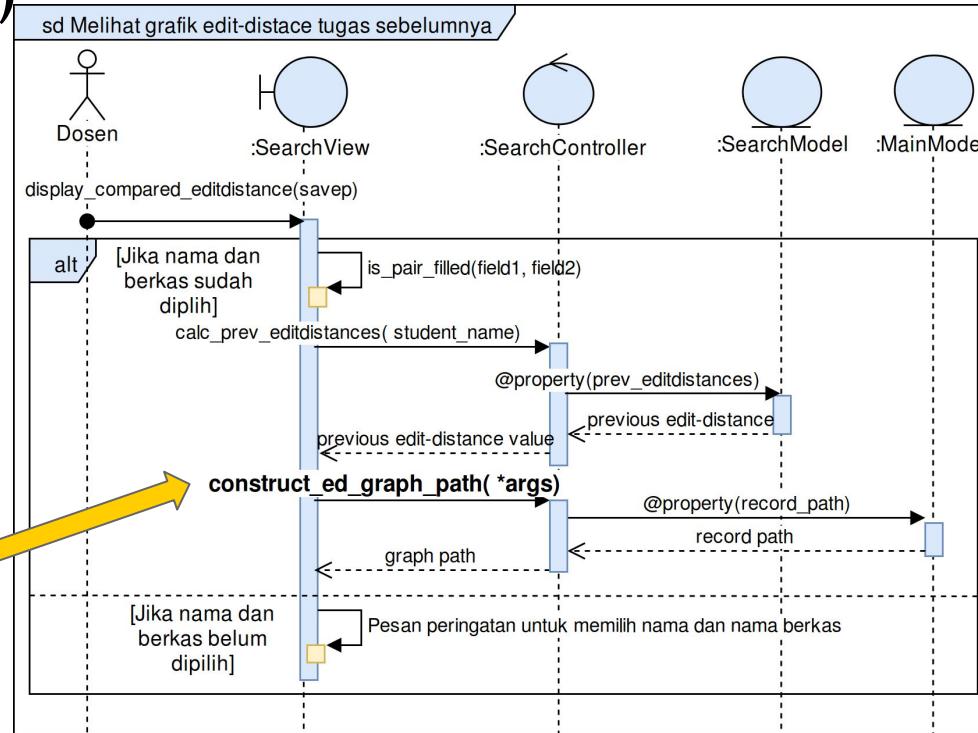




Hasil Perancangan Arsitektur



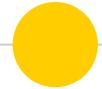
Hasil Perancangan Arsitektur (II)



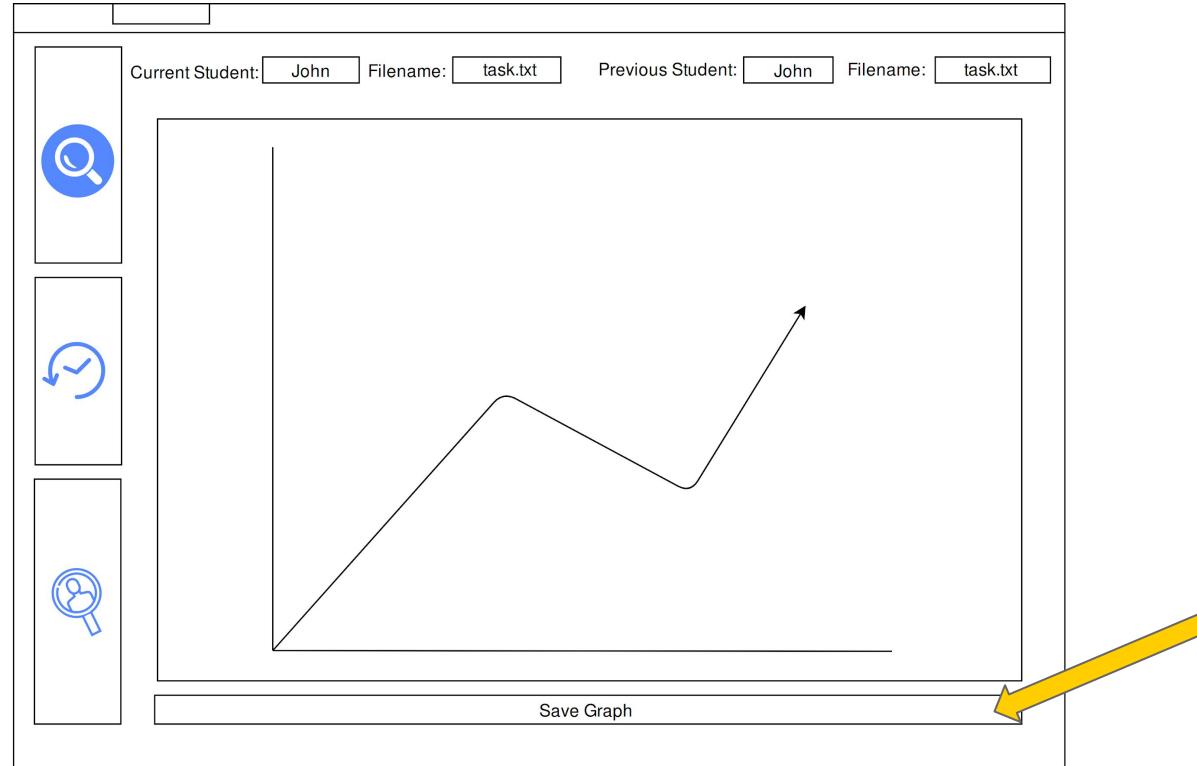


Hasil Perancangan Komponen

No	construct_ed_graph_path
1	START
2	CALL main_model RETURNING record_path
3	
4	IF passed argument == 1
5	SET graph_fmt to argument + ".png"
6	ELSE
7	SET graph_fmt to argument + " " + ".png"
8	ENDIF
9	SET graph_path record_path + "lupv_notes" + graph_fmt
10	RETURN graph_path
11	END

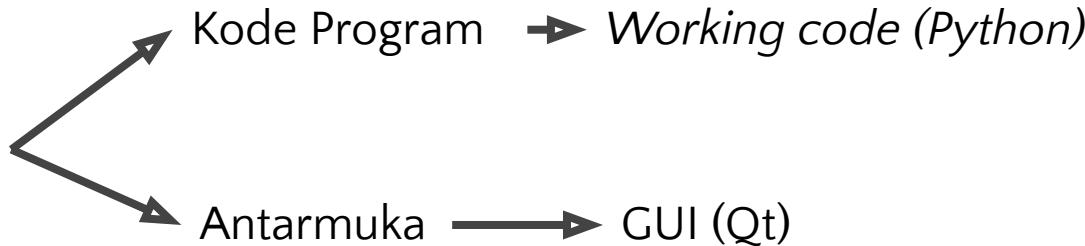


Hasil Perancangan Antarmuka





Implementasi



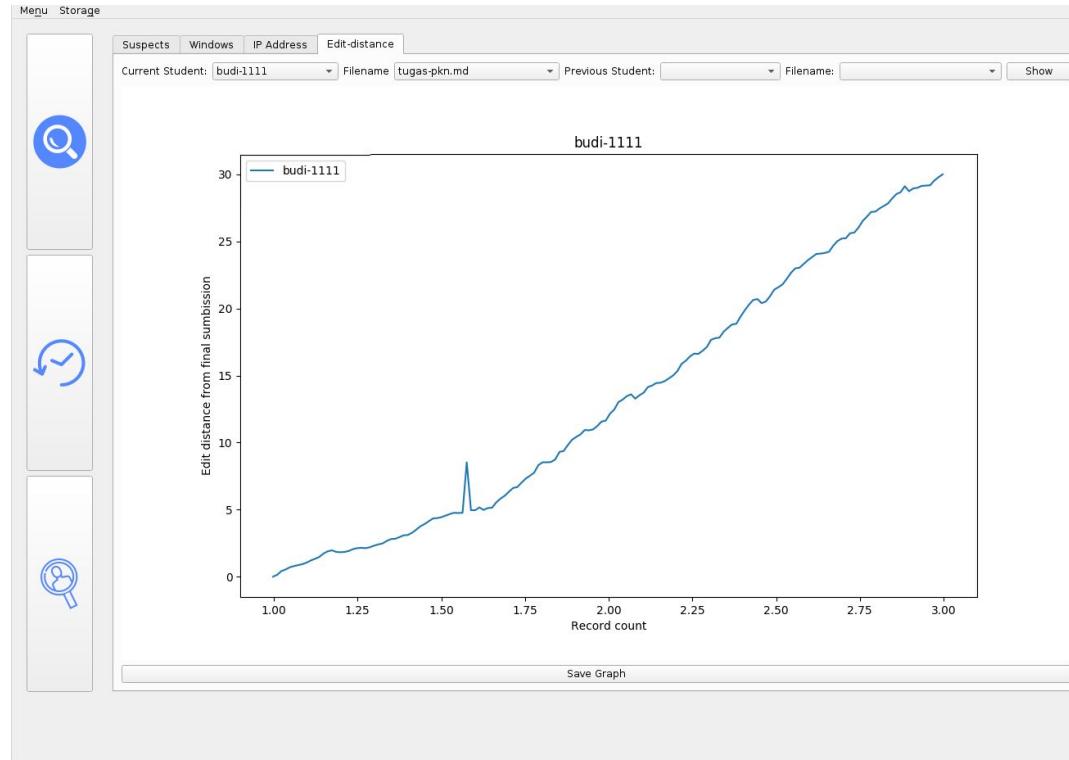


Hasil Implementasi Komponen

No	construct_ed_graph_path
1	<pre>def construct_ed_graph_path(self, *args): record_path = self._main_model.record_path if len(args) == 1: graph_fmt = "{}.png".format(args[0]) else: names = "_".join(args) graph_fmt = "{}.png".format(names) graph_path = join(record_path, "lupv-notes", graph_fmt) return graph_path</pre>

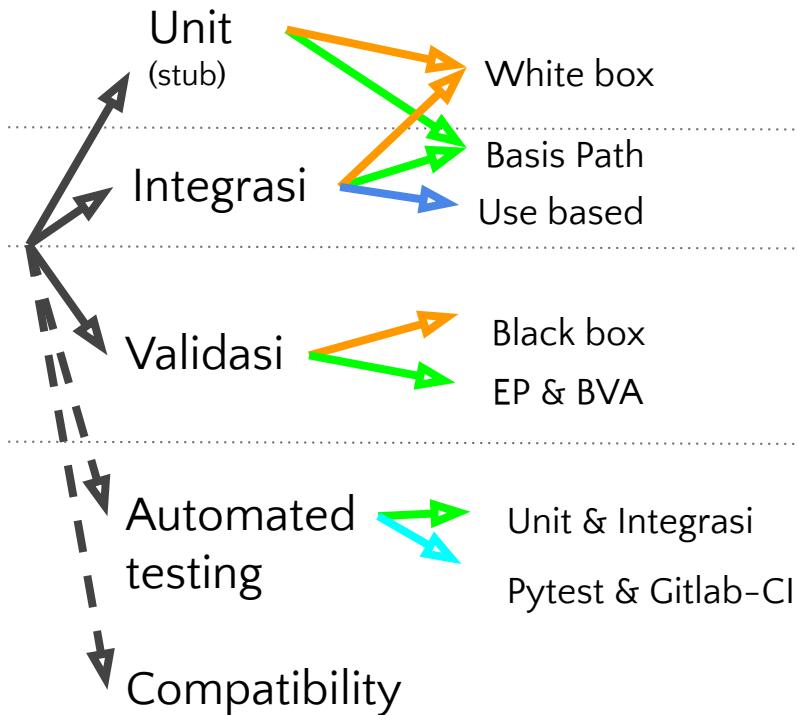


Hasil Implementasi Antarmuka





Pengujian



Implementasi kode ↔ Perancangan Komponen

Unit ↔ Unit

Sistem ↔ Kebutuhan



Pengujian

Unit
(stub)



- Metode: White box
- Teknik: Basis Path
- Hasil : **Valid + 100% code coverage**

Integrasi



- Metode: White box
- Teknik: Basis Path
- Pendekatan: Use based
- Hasil: **Valid**

Validasi



- Metode: Black box
- Teknik: EP & BVA
- Hasil: **Valid**



Hasil Pengujian Unit

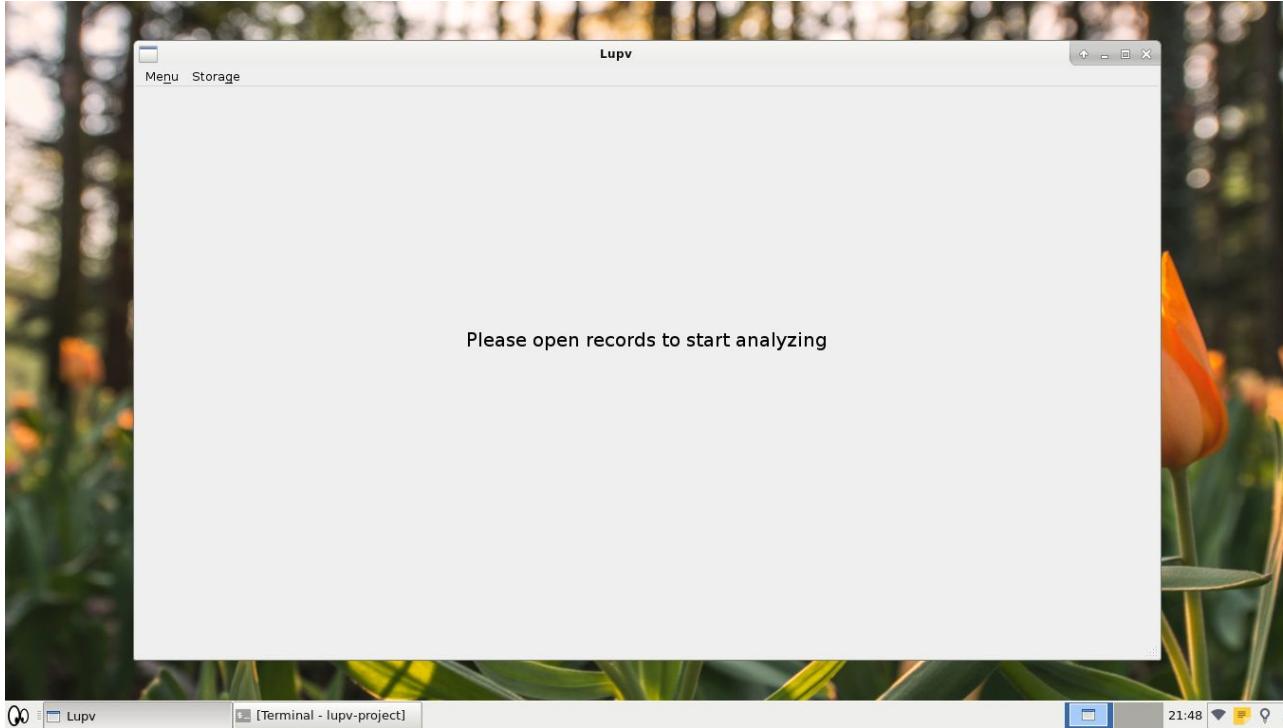
Name	Stmts	Miss	Cover
<hr/>			
Lupv/__init__.py	0	0	100%
Lupv/controllers/__init__.py	0	0	100%
Lupv/controllers/log.py	64	0	100%
Lupv/controllers/main.py	54	0	100%
Lupv/controllers/search.py	172	0	100%
Lupv/models/__init__.py	0	0	100%
Lupv/models/logs.py	75	0	100%
Lupv/models/main.py	42	0	100%
Lupv/models/search.py	30	0	100%
<hr/>			
TOTAL	437	0	100%
===== 91 passed in 6.45 seconds			

Hasil Automated Testing

```
tests/test_search_controller.py::TestSearchController::test_student_directories_iterator PASSED
tests/test_search_controller.py::TestSearchController::test_record_iterator PASSED
tests/test_search_controller.py::TestSearchController::test_analyze_suspects PASSED
tests/test_search_controller.py::TestSearchController::test_group_by_name PASSED
tests/test_search_controller.py::TestSearchController::test_get_suspects PASSED
tests/test_search_controller.py::TestSearchController::test_get_student_ips PASSED
tests/test_search_controller.py::TestSearchController::test_group_by_ip PASSED
tests/test_search_controller.py::TestSearchController::test_multigroup_child PASSED
tests/test_search_controller.py::TestSearchController::test_get_student_ip_groups PASSED
tests/test_search_controller.py::TestSearchController::test_idx_of_substring PASSED
tests/test_search_controller.py::TestSearchController::test_collect_student_windows PASSED
tests/test_search_controller.py::TestSearchController::test_get_student_windows PASSED
tests/test_search_controller.py::TestSearchController::test_populate_student_dirs PASSED
tests/test_search_controller.py::TestSearchController::test_load_prev_edidistances PASSED
tests/test_search_controller.py::TestSearchController::test_get_prev_student_names PASSED
tests/test_search_controller.py::TestSearchController::test_get_prev_filename_sample PASSED
tests/test_search_controller.py::TestSearchController::test_calc_prev_edidistances PASSED
tests/test_search_controller.py::TestSearchController::test_get_student_records PASSED
tests/test_search_controller.py::TestSearchController::test_calc_edidistances PASSED
tests/test_search_controller.py::TestSearchController::test_create_lupvnctes_dir PASSED
tests/test_search_controller.py::TestSearchController::test_construct_ed_graph_path PASSED
tests/test_search_controller.py::TestSearchController::test_construct_editistance_path PASSED
tests/test_search_controller.py::TestSearchController::test_export_edidistance PASSED
```

```
===== 91 passed in 8.62 seconds =====
```

Hasil Compatibility Testing





Kesimpulan

Rekayasa
Kebutuhan



- 2 **Aktor**
- 2 **Kebutuhan** untuk *Lup Recorder*
- 10 **Kebutuhan** untuk *Lup Viewer*
- 1 **Kebutuhan** non-fungsional

Perancangan
&
Implementasi



- 6 **class** untuk *Lup Recorder*
- 12 **class** untuk *Lup Viewer*
- *Pseudocode* → *Working code (Python)*
- Gambaran dasar → *GUI (Qt)*

Pengujian



- Unit, integrasi, validasi, *automated, compatibility.*
- Semua bernilai **valid**



Saran

- Tidak menyimpan data secara lokal, melainkan menggunakan penyimpanan server.
- Menambah sistem operasi yang didukung dengan melakukan *porting*.

Demo Aplikasi





Terima Kasih!

Plagiarisme

Penggunaan kata atau ide dari sebuah sumber tanpa menyertakan pengakuan.

(Meuschke & Gipp, 2013)





Kelemahan Plagiarism Detection

- Paradigma **polisi-kriminal** (Howard, 2002)
- Pola pikir **nilai oriented** (Dweck & Leggett, 1988)
- Mencari **cara-cara baru** agar tidak terdeteksi PDS (Kiss, 2013)
- Tidak dapat melacak **bantuan eksternal** (Hellas, Leinonen, & Ihantola, 2017)



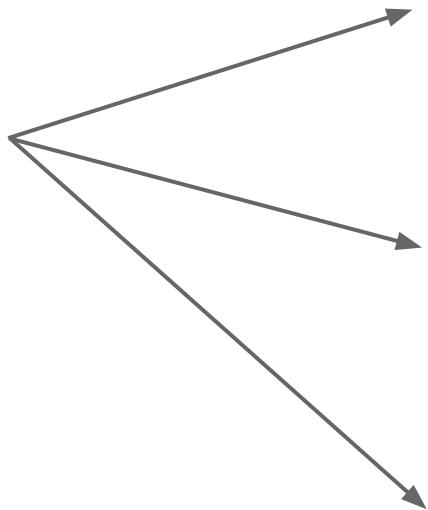
Kelemahan Plagiarism Detection (I)

- Lemah dalam:
 - Dokumen **terjemahan** (Meuschke & Gipp, 2013)
 - Berbagai **gaya sitiran** (Modiba, Pieterse, & Haskins, 2016)
 - Berbagai **bahasa** (Martins, et al., 2014)
- **Harga** tidak terjangkau (Agustina & Raharjo, 2017; Modiba, Pieterse, & Haskins, 2016)
- **Pelanggaran hak intelektual** dan hak milik (Park, 2004)



Plagiarism Prevention

Metode
Plagiarism
Prevention



Formative assessment pada siswa (Leung & Cheng, 2017)

Edukasi siswa tentang plagiarisme (McLafferty & Foust, 2004)

Pencegahan dengan perangkat lunak (Hellas, Leinonen, & Ihantola, 2017)



Kelemahan

Mengabaikan proses penggerjaan



Membeli tugas
(Leung & Cheng, 2017)

Analisis hanya di akhir



Kecurangan selama mengerjakan
(Hellas, Leinonen, & Ihantola, 2017)

Tidak memiliki *log*



Tidak dapat menilai usaha
(Hellas, Leinonen, & Ihantola, 2017)

Terikat suatu *platform*



Bentuk tugas & *software* terbatas



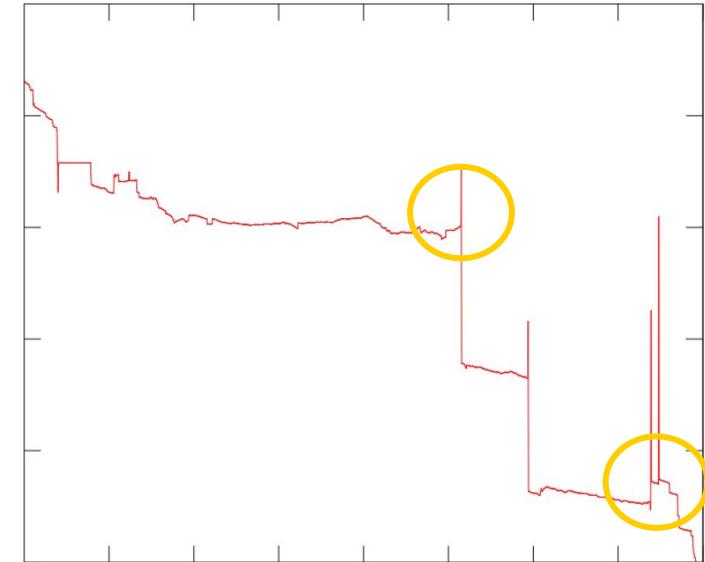
Batasan

- Format yang didukung hanya *plain text*
- Sementara hanya berjalan di **GNU/Linux**
- Pengolah kata harus memiliki fitur *autosave*



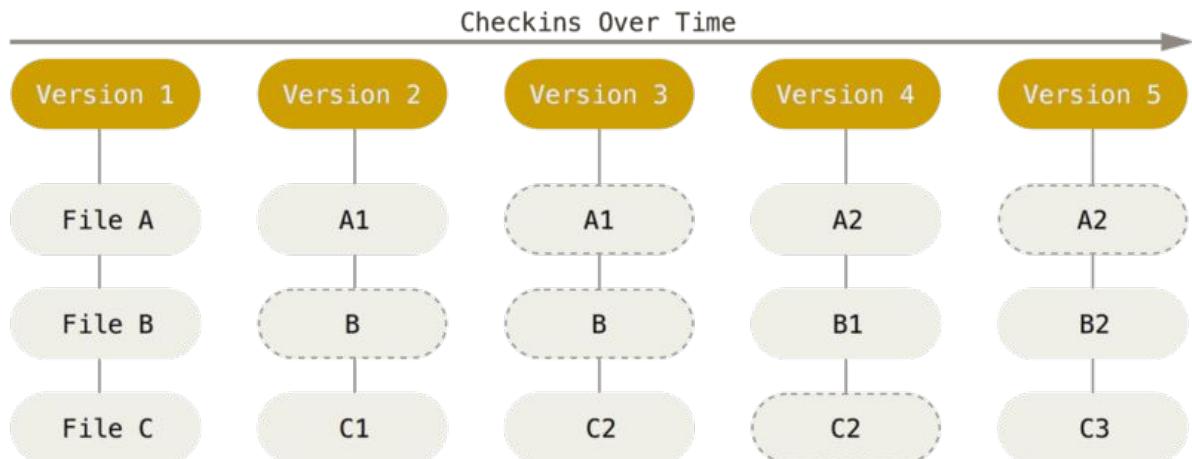
Algoritme Edit-distance

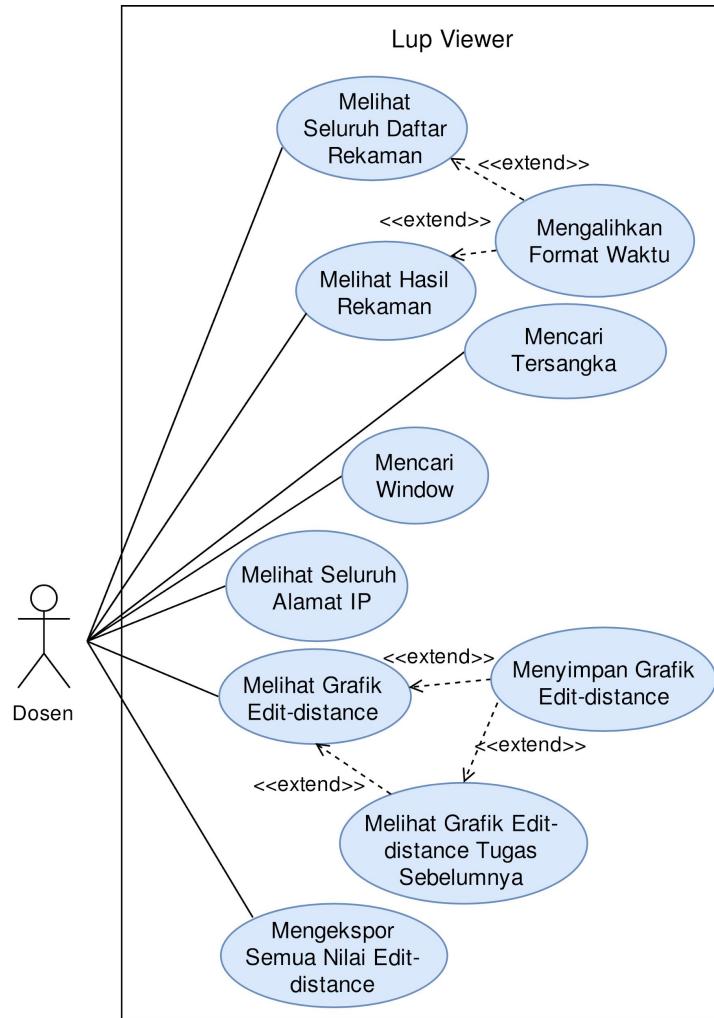
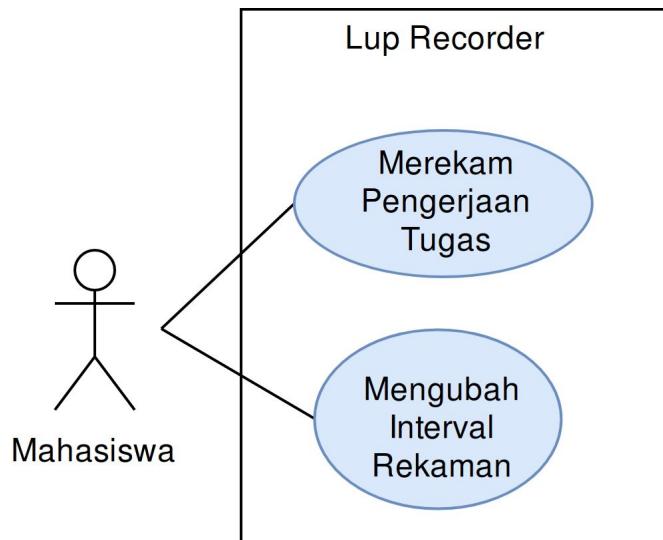
- 1) **tekkom** → fekkom (“t” → “f”)
- 2) fekkom → fíkkom (“e” → “í”)
- 3) fíkkom → **fílkom** (“k” → “l”)





Teknologi yang digunakan





When to Use Use Cases

83

83

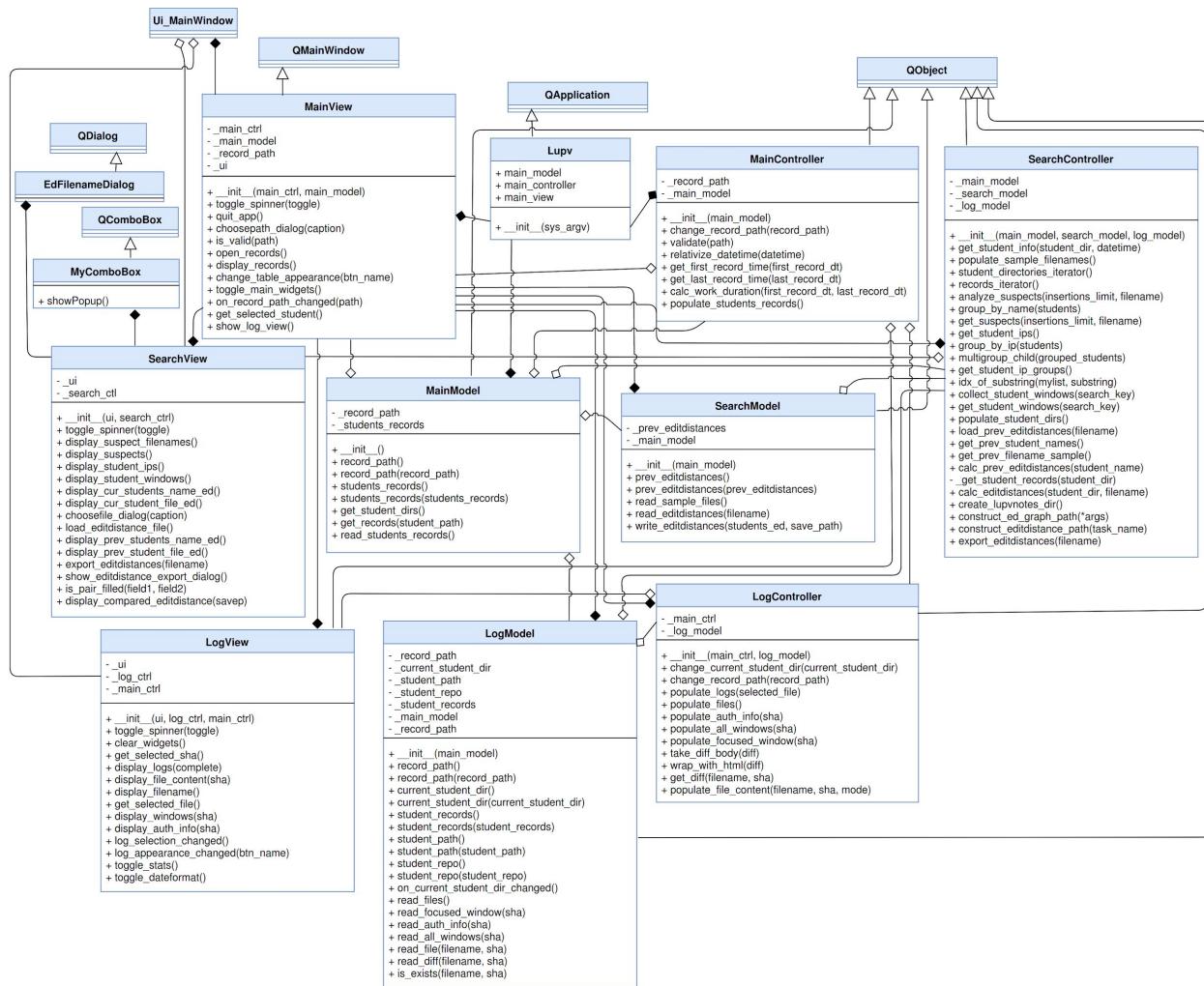
Use cases are a valuable tool to help understand the functional requirements of a system. A first pass at use cases should be made early on. More detailed versions of use cases should be worked just prior to developing that use case.

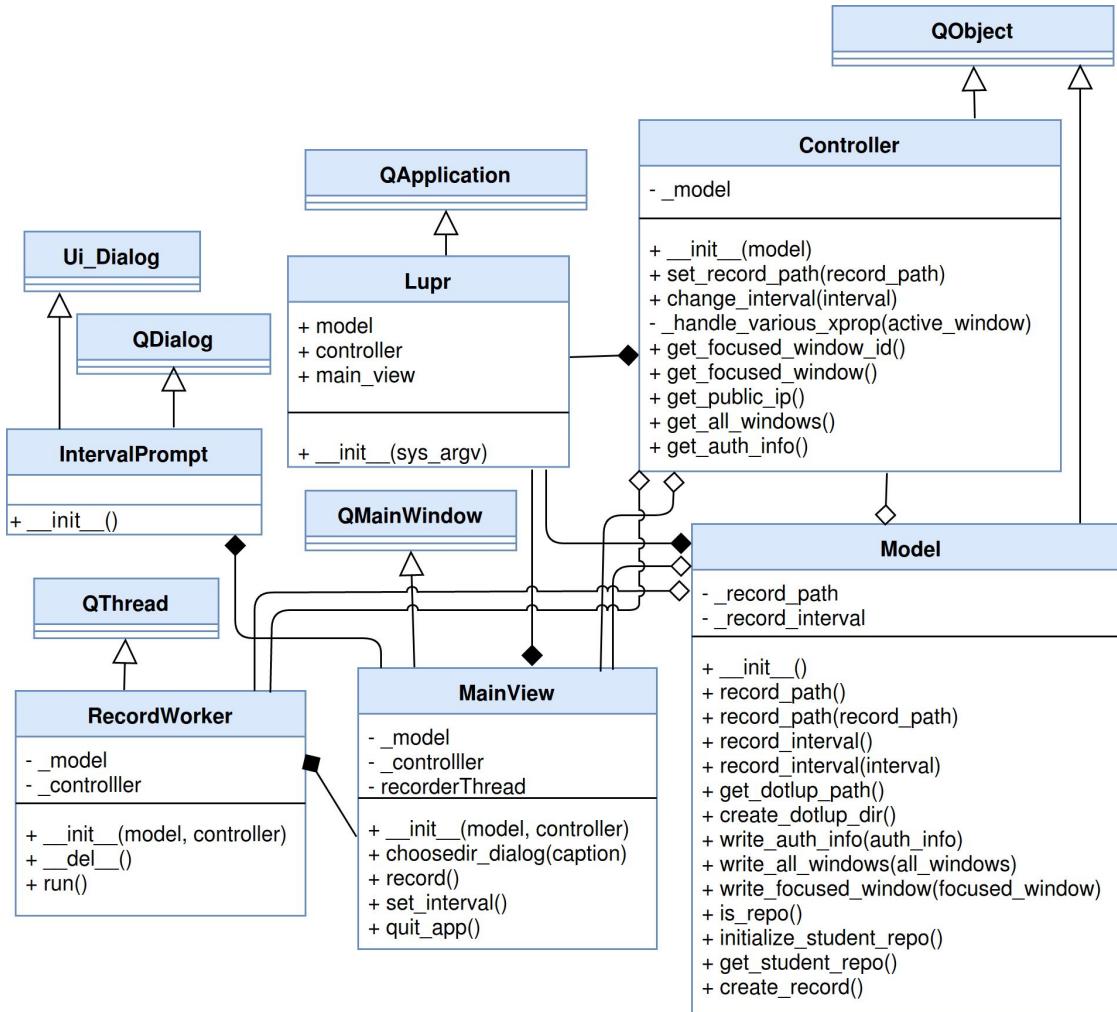
It is important to remember that use cases represent an *external* view of the system. As such, don't expect any correlations between use cases and the classes inside the system.

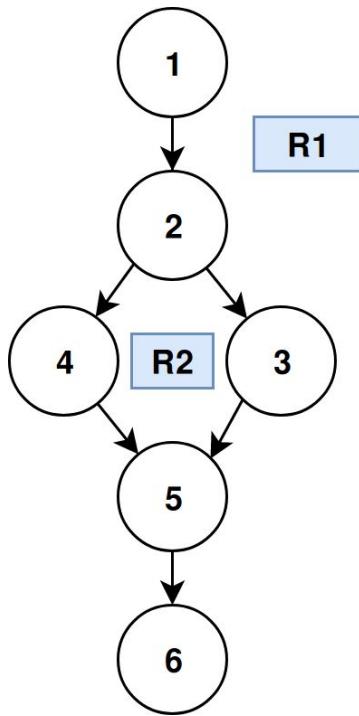
Flow of events untuk Menyimpan Grafik *Edit-distance*

Kode Kebutuhan	LUPV-F-08
Objective	Dosen dapat menyimpan grafik <i>edit-distance</i> .
Actor	Dosen
Pre-condition	Grafik <i>edit-distance</i> telah tersedia.
Main flow	<ol style="list-style-type: none">1. Dosen memilih tombol “Save Graph”.2. Sistem menyimpan grafik pada direktori “lupv-notes” pada direktori tugas.3. Sistem menampilkan pesan informasi “Graph saved to «lokasi berkas»” menandakan bahwa grafik telah disimpan.
Alternative flows	-
Post-condition	Grafik <i>edit distance</i> disimpan oleh sistem.

Main flow	<ol style="list-style-type: none"> 1. Dosen memilih tombol “Open Records”. 2. Sistem menampilkan dialog untuk memilih direktori rekaman yang akan dibuka. Pada dialog tersebut terdapat tombol “Choose” dan “Cancel”. 3. Dosen memilih tombol “Choose”. 4. Sistem melakukan validasi direktori rekaman dan menampilkan seluruh daftar rekaman mahasiswa. Data yang ditampilkan meliputi nama, nim, total rekaman, waktu rekaman awal, waktu rekaman akhir, dan durasi penggerjaan.
Alternative flows	<ol style="list-style-type: none"> 3.1 Apabila dosen memilih tombol “Cancel”, maka berkas rekaman tidak dibuka. 4.1 Apabila direktori tidak valid, yaitu tidak sesuai dengan spesifikasi kebutuhan LUPV-F-01-02 dan LUPV-F-01-03, maka sistem menampilkan pesan peringatan “<i>Not a valid Task directory (baris baru)</i> <i>Contains invalid Task: «direktori yang tidak valid»</i>”.
	<ol style="list-style-type: none"> 4.2 Perluasan ke <i>use case</i> dengan kode kebutuhan LUPV-F-10.
Post-condition	Seluruh daftar rekaman ditampilkan oleh sistem.







Jalur	Prosedur Uji	Expected Result	Result	Status
1	<p><i>Class Driver</i></p> <p><i>TestSearchController</i> memanggil function <i>construct_ed_graph_path</i> dengan argument “ani-1111”.</p>	<p><i>Return value function construct_ed_graph_path bernilai “/lupv-notes/ani-1111.png”.</i></p>	As expected	Valid
2	<p><i>Class Driver</i></p> <p><i>TestSearchController</i> memanggil function <i>construct_ed_graph_path</i> dengan argument “ani-1111”, “budi-2222”.</p>	<p><i>Return value function construct_ed_graph_path bernilai “/lupv-notes/ani-1111_budi-2222.png”.</i></p>	As expected	Valid

No

construct_ed_graph_path

```
1 def test_construct_ed_graph_path(self, search_ctrl):
2     """Test constructing editdistance graph path."""
3     graph_path = search_ctrl.construct_ed_graph_path(
4         "ani-1111")
5     graph_path_2 = search_ctrl.construct_ed_graph_path(
6         "ani-1111", "budi-2222")
7
8     assert "/lupv-notes/ani-1111.png" in graph_path
9     assert "/lupv-notes/ani-1111_budi-2222.png" in
    →   graph_path_2
```



 passedPipeline #68021478 triggered 5 days ago by  azzamsa

Reword "focused" -> "focused_window"

to make it more readable

⌚ 1 job for master in 1 minute and 37 seconds

 latest→ a1d3bd32  **Pipeline** Jobs 1**Test** pytest

GitLab Projects Groups Activity Milestones Snippets

Luvv

Project Repository Issues Merge Requests

CI / CD Pipelines Jobs Schedules Charts

Operations Registry Packages Wiki Snippets Settings

```
===== slowest 10 test durations =====
0.18s setup tests/test_controller.py::TestMainController::test_validate_invalid_dir
0.15s call tests/test_integration.py::TestIntegration::test_populate_logs
0.13s setup tests/test_search_controller.py::TestSearchController::test_export_edidistance
0.11s setup tests/test_search_controller.py::TestSearchController::test_create_lupvnotes_dir
0.10s setup tests/test_controller.py::TestLogController::test_take_diff_body
0.10s setup tests/test_controller.py::TestMainController::test_validate_exception
0.10s setup tests/test_controller.py::TestMainController::test_validate_invalid_dir_name
0.05s call tests/test_search_controller.py::TestSearchController::test_calc_edidistances
0.03s setup tests/test_log_view.py::TestLogView::test_toggle_spinner
0.03s call tests/test_main_view.py::TestMainView::test_show_log_view
===== 91 passed in 3.95 seconds =====
$ codecov --token=9eb87afa-bfbc-4826-b3b0-d2c2950fe836
```



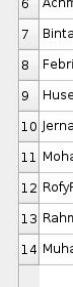
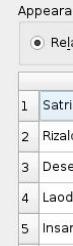
v2.0.15

```
==> Detecting CI provider
Gitlab CI Detected
==> Preparing upload
==> Processing gcov (disable by -X gcov)
  Executing gcov (find /builds/azzamsa/lupv -not -path './bower_components/**' -not -path './node_modules/**' -name '*.gcov' | xargs gcov)
==> Collecting reports
  + /builds/azzamsa/lupv/pytestvenv/bin/coverage3 bytes=246
  + /builds/azzamsa/lupv/pytestvenv/bin/coverage bytes=246
  + /builds/azzamsa/lupv/pytestvenv/bin/coverage-3.6 bytes=246
  Generating coverage xml reports for Python
  + /builds/azzamsa/lupv/coverage.xml bytes=36492
==> Uploading
  .url https://codecov.io
  .query commit=ald3bd32ebb9597921cdfd98eb2f7fb11304865&branch=master&token=<secret>&service=gitlab
  Pinging Codecov...
  Uploading to S3...
https://codecov.io/gitlab/azzamsa/lupv/commit/ald3bd32ebb9597921cdfd98eb2f7fb11304865
```

Two yellow arrows point from the right side of the image towards the bottom of the terminal output, highlighting the command '\$ codecov --token=9eb87afa-bfbc-4826-b3b0-d2c2950fe836' and the resulting URL 'https://codecov.io/gitlab/azzamsa/lupv/commit/ald3bd32ebb9597921cdfd98eb2f7fb11304865'.

No	Nama Penguji	Desktop Environment	Versi
1	Bintang Dimas PAR	GNOME	3.28.2
2	Muhammad Iqbal K	GNOME	3.28.2
3	Laode Muhamad Fauzan	KDE	5.9.15
4	Husein Abdulbar	KDE	5.15.5
5	Rofy Firmansyah R	KDE	5.15.5
6	Satria Adhi Kharisma	XFCE	4.12
7	Dese Narfa Firmansyah	XFCE	4.12
8	Rizaldy Firmansyah Y	Unity	7
9	Insan Nurzaman	Cinnamon	3.6.6
10	Jerna Ferda Kusuma	Cinnamon	4.0.10
11	Rahmat Guntur Husodo	Cinnamon	4.0.10
12	Febristya AF	Pantheon	5.0
13	Achmad Rizki Aditama	OpenBox	3.6.1
14	Mohammad Anton RS	i3wm	4.16.1

Menu Storage



Appearance						
		<input checked="" type="radio"/> Relative DateTime	<input type="radio"/> Real DateTime			
	Name	Student ID	Total Records	First Record	Last Record	Work Duration
1	SatriaAdhiKharisma	155150207111119	193	Thu, 21 Mar 2019, 20:37:00	Thu, 21 Mar 2019, 20:49:54	0:12:54
2	RizaldyFirmansyahYulianto	155150207111112	273	Fri, 22 Mar 2019, 01:27:01	Fri, 22 Mar 2019, 03:42:27	2:15:26
3	DeseNarfaFirmansyah	1551502011111153	527	Thu, 21 Mar 2019, 18:44:43	Thu, 21 Mar 2019, 19:23:30	0:38:47
4	LaodeMuhamadFauzan	1551502001111158	146	Wed, 20 Mar 2019, 16:35:02	Wed, 20 Mar 2019, 17:06:15	0:31:13
5	InsanNurzaman	165150200111033	300	Wed, 20 Mar 2019, 19:38:20	Wed, 20 Mar 2019, 20:03:24	0:25:04
6	AchmadRizkiAditama	155150207111158	344	Wed, 20 Mar 2019, 20:58:02	Wed, 20 Mar 2019, 21:21:33	0:23:31
7	BintangDimasPAR	1551502011111347	373	Wed, 20 Mar 2019, 22:56:24	Wed, 20 Mar 2019, 23:35:33	0:39:09
8	FebristyaAF	155150201111126	197	Thu, 21 Mar 2019, 20:04:01	Thu, 21 Mar 2019, 20:20:24	0:16:23
9	HuseinAbdulbar	155150200111193	177	Fri, 22 Mar 2019, 08:07:48	Fri, 22 Mar 2019, 08:18:00	0:10:12
10	JernaFerdaKusuma	155080200111053	148	Sat, 23 Mar 2019, 12:55:03	Sat, 23 Mar 2019, 13:15:50	0:20:47
11	MohammadAntonRizkiSyahputra	155150200111104	240	Thu, 21 Mar 2019, 20:04:03	Thu, 21 Mar 2019, 20:24:27	0:20:24
12	RofyFirmansyahR	1551502001111115	270	Thu, 21 Mar 2019, 21:17:22	Thu, 21 Mar 2019, 21:32:39	0:15:17
13	RahmatGunturHusodo	165150201111282	335	Thu, 21 Mar 2019, 18:28:21	Thu, 21 Mar 2019, 18:56:18	0:27:57
14	MuhammadIqbalKurniawan	155150200111259	385	Wed, 20 Mar 2019, 19:44:08	Wed, 20 Mar 2019, 20:12:24	0:28:16

Aggregation (light)

```
class B(object): pass
```

```
class A(object):
```

```
    def __init__(self, b):  
        self.b = b
```

```
    b = B()
```

```
    a = A(b)
```

Pass A to B (as arguments)



Composition (bold)

```
class A(object):
```

```
    def __init__(self):
```

```
        self.b = B()
```

Create B inside A

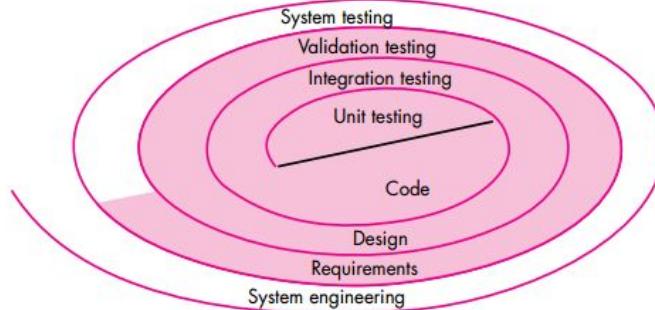


Stub

```
Assert "123" = return_value
```

Mock

```
Assert "123" = fake.return_value
```

FIGURE 17.1**Testing strategy****468****PART THREE** QUALITY MANAGEMENT

software, and WebApps disappears. Testing focuses on user-visible actions and user-recognizable output from the system.

KEY POINT

Like all other testing steps, validation tries to uncover errors, but the focus is at the requirements level—on things that will be immediately apparent to the end user.

Validation can be defined in many ways, but a simple (albeit harsh) definition is that validation succeeds when software functions in a manner that can be reasonably expected by the customer. At this point a battle-hardened software developer might protest: “Who or what is the arbiter of reasonable expectations?” If a *Software Requirements Specification* has been developed, it describes all user-visible attributes of the software and contains a *Validation Criteria* section that forms the basis for a validation-testing approach.

17.6.1 Validation-Test Criteria

more detail in Chapter 22.

"Given enough eyeballs, all bugs are shallow (e.g., given a large enough beta-tester and co-developer base, almost every problem will be characterized quickly and the fix obvious to someone)."

E. Raymond

17.6.3 Alpha and Beta Testing

468

It is virtually impossible for a software developer to foresee how the customer will really use a program. Instructions for use may be misinterpreted; strange combinations of data may be regularly used; output that seemed clear to the tester may be unintelligible to a user in the field.

When custom software is built for one customer, a series of acceptance tests are conducted to enable the customer to validate all requirements. Conducted by the end user rather than software engineers, an acceptance test can range from an informal "test drive" to a planned and systematically executed series of tests. In fact, acceptance testing can be conducted over a period of weeks or months, thereby uncovering cumulative errors that might degrade the system over time.

Donate to the Python Software Foundation or Purchase a PyCharm License to Benefit the PSF! [Donate Now ↗](#)



Search projects



Help

Donate

Log in

Register

Lupv 1.6.37

pip install Lupv



Latest version

Last released: Mar 20, 2019

Lup viewer

Navigation

☰ Project description

⟳ Release history

⬇ Download files

Project links

⌂ Homepage

Statistics

Project description

Lupv - Lup Viewer

[pypi v1.6.37](#) [license GPLv3](#) [python 3.6](#)

Screenshot

lupv-thumb

Requirements

See [requirements.txt](#)

Statistics

View statistics for this project via
[Libraries.io](#), or by using [Google BigQuery](#).

Meta

License: MIT License (GPLv3)

Author: [azzamsa](#)

Maintainers



[azzamsa](#)

Classifiers

GPLv3

License

[OSI Approved :: MIT License](#)

Programming Language

[Python :: 3](#)

[Python :: 3.6](#)

[Python :: Implementation ::](#)

[CPython](#)

See [requirements.txt](#)

Platform

- GNU/Linux

Installation

- Use binaries from releases tag.

Usage

- You can watch the screencast [here](#)

License

[STATUS](#)
[PROJECT](#)
[LAST COMMIT](#)
[ISSUES](#)


azzamsa 6 days ago
Reword "focused" -> "focused_window"

0	0	21
NEW	FIXED	TOTAL



azzamsa 18 days ago
Bump version to v1.5.6

0	0	22
NEW	FIXED	TOTAL

Lupv

Navigation

[Developer Convention](#)
[Main Controller](#)
[Log Controller](#)
[Search Controller](#)
[Main Model](#)
[Logs Model](#)
[Search Model](#)
[Main View](#)
[Log View](#)
[Search View](#)
[Test Model](#)
[Test Controller](#)
[Quick search](#)

Welcome to Lupv's documentation!

Analyze your student's records with Lupv

The Developer Convention

- [Developer Convention](#)
 - [Concept](#)
 - [Naming Variables](#)

The Code Documentation

- [Main Controller](#)
- [Log Controller](#)
- [Search Controller](#)
- [Main Model](#)
- [Logs Model](#)
- [Search Model](#)
- [Main View](#)
- [Log View](#)
- [Search View](#)
- [Test Model](#)
 - [Test Main Model](#)
 - [Test Log Model](#)
 - [Test Search Model](#)
- [Test Controller](#)
 - [Main Controller](#)
 - [Log Controller](#)
 - [Search Controller](#)

Lupv

Navigation

[Developer Convention](#)
[Main Controller](#)
[Log Controller](#)
[Search Controller](#)
[Main Model](#)
[Logs Model](#)
[Search Model](#)
[Main View](#)
[Log View](#)
[Search View](#)
[Test Model](#)
[Test Controller](#)

Quick search

Go

Main Controller

```
class Lupv.controllers.main.MainController(main_model)
```

```
calc_work_duration(first_record_dt, last_record_dt)
```

Calculate duration between last and first record time.

```
change_record_path(record_path)
```

```
get_first_record_time(first_record_dt)
```

Take the first record time.

```
get_last_record_time(last_record_dt)
```

Take the last record time.

```
populate_students_records()
```

Return list of student records.

```
relativize_datetime(datetime)
```

Convert datetime into its relative version.

```
validate(path)
```

Check if path contain valid student directory.

This is necessary because invalid directory will break `read_records` and make application crash.



Links

- [Dokumen Skripsi](#)
- [Repository](#)
- [Youtube Demo](#)
- [PyPI](#) (pip install lupv)



Credits

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by [SlidesCarnival](#)
- Pictures by [FreePik](#)