

HASIL CEK_Implementasi Test Driven

by Tif Implementasi Test Driven

Submission date: 12-Aug-2023 11:41AM (UTC+0700)

Submission ID: 2144703669

File name: 12. Implementasi Test Driven Development Pada Pengembangan Aplikasi Android Untuk Mahasiswa Universitas Ahmad Dahlan.pdf (271.67K)

Word count: 3041

Character count: 18472

Implementasi Test Driven Development Pada Pengembangan Aplikasi Android Untuk Mahasiswa Universitas Ahmad Dahlan

Jamalludin¹, Herman Yuliansyah¹, Sri Winiati¹, Imam Riadi²

¹Program Studi Teknik Informatika, ²Magister Teknik Informatika
Universitas Ahmad Dahlan

^{1,2}Kampus 3, Jln. Prof.Dr. Supomo, Janturan, Yogyakarta 55164

e-mail : jamalludin1300018089@webmail.uad.ac.id, herman.yuliansyah@tif.uad.ac.id,
sri.winiarti@tif.uad.ac.id, imam.riadi@mti.uad.ac.id

Abstract

Test-driven development (TDD) is a software development process with a very short software development cycle repetition: first the developer writes the test code for defining the function, then produce the code to pass the test and finally the acceptable new code. The aim of this research is development android application with Test Driven Development approach that have feature like academic counseling, courses information, lecturer schedule, department schedule, classroom schedule for Universitas Ahmad Dahlan's students. The result of this research is the development of mobile application and the unit testing. It can show the software features have been through the testing and have been appropriate.

Keywords: *Android application; Courses schedule; Test driven development*

Abstrak

*Test-driven development (TDD) merupakan suatu proses pengembangan perangkat lunak dengan pengulangan siklus pengembangan yang sangat singkat: pertama pengembang menulis kode pengujian untuk pendefinisian fungsi yang diinginkan, kemudian menghasilkan kode program untuk lulus tes dan akhirnya kode baru yang dapat diterima. Tujuan penelitian ini adalah pengembangan aplikasi mobile dengan menggunakan pendekatan *Test Driven Development* yang dapat melakukan kegiatan perwalian dan memberikan informasi perkuliahan, jadwal dosen, jadwal program studi, dan jadwal ruang untuk mahasiswa Universitas Ahmad Dahlan. Hasil dari penelitian ini adalah pengembangan aplikasi mobile dan unit testingnya sehingga dapat menunjukkan fitur perangkat lunak telah melalui pengujian dan telah sesuai.*

Kata kunci: *Aplikasi android; Penjadwalan perkuliahan; Test driven development*

1. Pendahuluan

Universitas Ahmad Dahlan (UAD) adalah salah satu universitas swasta terkemuka di Propinsi D.I. Yogyakarta. UAD memiliki 10 (Sepuluh) fakultas dengan total program studi sebanyak 34 (tiga puluh empat) program studi. Fakultas Teknologi Industri (FTI) UAD memiliki beberapa sistem akademik yang menunjang proses akademika. Sistem tersebut antara lain portal.uad.ac.id sebagai Sistem Informasi Akademik, sistem perwalian.fti.uad.ac.id sebagai sistem Perwalian akademik, dan simeru.uad.ac.id sebagai Sistem Manajemen Ruang (SIMERU) yang dapat diakses melalui web secara *online*. Sistem perwalian.fti.uad.ac.id adalah sistem yang digunakan oleh mahasiswa untuk merekam data perwalian yang telah dilakukan antara dosen pembimbing akademik dengan mahasiswa. Perwalian adalah sebuah proses yang dilakukan oleh mahasiswa ke dosen pembimbing akademik untuk mendapatkan saran dan nasehat yang berhubungan dengan kegiatan akademik [1]. Dosen Pembimbing akademik bertugas memantau

perkembangan prestasi mahasiswa, membantu dalam penyusunan perkuliahan, dan memberikan konsultasi dan arahan kepada mahasiswa baik dalam bidang akademik maupun non akademik [2].

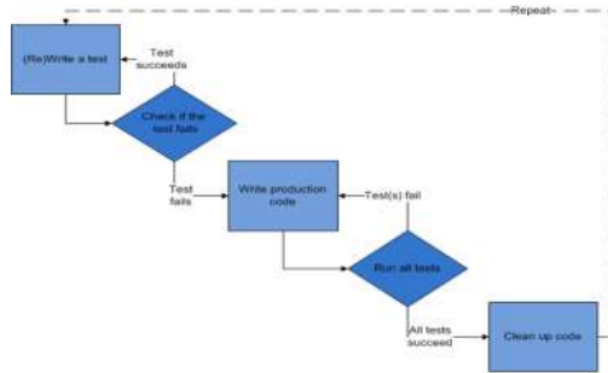
SIMERU yang beralamat di <http://simeru.uad.ac.id> merupakan sistem yang berfungsi sebagai pusat informasi jadwal ruang, jadwal dosen, dan jadwal perkuliahan di UAD. Berdasarkan hasil wawancara dengan petugas pengelola SIMERU FTI UAD mengatakan bahwa informasi yang terdapat di SIMERU masih kurang efektif dalam penyampaian informasi ke pada mahasiswa baik informasi mata kuliah kosong maupun informasi perpindahan ruang perkuliahan, karena informasi hanya disebar melalui papan pengumuman di kantor simeru dan di ruang kelas perkuliahan dengan menggunakan kertas, dikarenakan sistem simeru saat ini belum adanya notifikasi jika perkuliahan kosong maupun perpindahan ruang perkuliahan, sehingga informasi tidak dapat dilihat oleh semua mahasiswa yang bersangkutan dengan matakuliah tersebut. Beberapa penelitian terdahulu telah dilakukan terkait pengembangan aplikasi android untuk informasi jadwal perkuliahan [3][4][5][6][7][8][9], pengembangan aplikasi web untuk informasi jadwal perkuliahan [10][11][12][13], pengembangan aplikasi desktop untuk informasi jadwal perkuliahan [14]. Selain informasi jadwal perkuliahan juga telah dilakukan penelitian terkait pengembangan aplikasi perwalian sebagai media komunikasi antara dosen dan mahasiswa terkait perkembangan akademik [11][15] dan sistem informasi peminjaman penggunaan ruangan [16].

Penelitian ini akan mengimplementasikan *Test Driven Development* pada pengembangan aplikasi di android. Hal ini dikarenakan bahwa aplikasi android merupakan aplikasi mobile yang populer dan telah banyak dihasilkan beberapa aplikasi telah dimanfaatkan secara luas. Beberapa pemanfaatan aplikasi android diantaranya adalah aplikasi android pada perusahaan telkomunikasi untuk pencarian *distribution point* dengan memanfaatkan sistem layanan berbasis lokasi [17]. Aplikasi *mobile* sebagai sarana media promosi produk bagi usaha mikro kecil dan menengah serta menginformasikan lokasi layanan perbankan terdekat dari usaha tersebut [18]. Aplikasi android dapat digunakan juga sebagai aplikasi pemandu pada museum. Aplikasi ini memanfaatkan *QR Code* sebagai pengidentifikasi layanan pada museum Gunung Merapi (MGM) [19]. Selain sebagai pemandu di museum, aplikasi android juga telah digunakan untuk informasi pengenalan satwa di kebun binatang dengan memanfaatkan koordinat GPS dari kandang satwa yang dikunjungi oleh pengunjung [20]. Aplikasi android juga telah dimanfaatkan sebagai aplikasi *Point of Sales* pada kafe untuk kasir *portable* atau tidak menetap sesuai dengan posisi duduk dari para pembeli di kafe tersebut. Sehingga pembeli dapat langsung melakukan pembayaran saat selesai bertransaksi. Aplikasi ini juga dihubungkan dengan *bluetooth printer portable* sebagai media cetak [21].

Berdasarkan wawancara dengan narasumber dosen dan mahasiswa ditemukan permasalahan-permasalahan yaitu yang terjadi mahasiswa kesulitan menemui dosen pembimbing, kesulitan mengetahui informasi perkuliahan seperti kuliah kosong, perpindahan ruang kuliah, mahasiswa harus mengecek ruang kelas atau kantor SIMERU untuk mengetahui informasi tersebut, tidak adanya fitur notifikasi di sistem simeru ketika ada informasi tentang perkuliahan. Dari pengolahan data kuisoiner sebanyak 25 responden dari 30 responden bahwa aplikasi simeru dan perwalian setuju untuk dijadikan sebuah aplikasi mobile. Maka untuk itu perlu dibuat sebuah aplikasi android untuk informasi ruang dan perwalian. Pengembangan aplikasi dilakukan dengan metode TDD serta melakukan pengujian Usability untuk mengetahui tingkat penerimaan pengguna dengan menggunakan metode *System Usability Scale* (SUS).

2. Metode Penelitian

Metode dalam penelitian ini adalah dengan menggunakan pendekatan *Test Driven Development* (TDD) [22]. TDD adalah proses pengembangan perangkat lunak yang bergantung pada pengulangan siklus pengembangan yang sangat singkat: pertama pengembang menulis kegagalan uji otomatis yang mendefinisikan peningkatan yang diinginkan atau fungsi baru, dan kemudian menghasilkan kode untuk lulus tes dan akhirnya refraktor kode baru untuk standar yang dapat diterima. TDD adalah pendekatan yang baik untuk pengembangan perangkat lunak baru tetapi lebih memakan waktu proses model ketika menguji sistem perangkat lunak yang ada. Alur kerja dari TDD dapat dilihat pada Gambar 1. Pada Gambar 1 merupakan alur kerja dari *Test Driven Development* dimana pada tahapan awal adalah membuat *unit testing*, kemudian *unit test* dijalankan jika hasil menunjukkan sukses maka *unit testing* perlu diperbaiki jika hasil menunjukkan *fails* maka dilanjutkan dengan membuat kode program. Kode program dinyatakan sudah selesai jika serangkaian pengujian kode program telah berhasil diujikan dengan *unit testing* yang telah dibuat.



Gambar 1. Alur Kerja *Test driven development* [22].

3. Hasil dan Pembahasan

3.1. *Testing Unit*

Pembuatan *unit testing* menggunakan JUnit yang merupakan *unit testing* yang sudah disediakan oleh android studio, kode program untuk melakukan pengujian dapat dilihat pada Gambar 2.

```

@Rule(MethodsManuallySetUp.class)
public class TestApilhas {

    String url = "http://localhost:8080/berita";
    Context context;

    @Before
    public void setUp() {
        context = InstrumentationRegistry.getContext();
    }

    @Test
    public void useTest() throws Exception {
        BeritaCallback bc = new BeritaCallback(context);
        final SettableFuture<String> ff = SettableFuture.create();
        bc.BeritaCallback("004107321", ff, bc);
        ff.set("success");
    }

    String txt = ff.get(1000, TimeUnit.SECONDS);
    assertEquals(txt, "004107321");
    }
    
```

Gambar 2. *Testing unit*

Pada Gambar 2 merupakan kode program untuk melakukan pengujian dengan menggunakan JUnit yang sudah disediakan oleh android studio, kode pengujian pada Gambar 2 digunakan untuk menguji data yang diambil dari web service sama dengan data yang ditampilkan pada aplikasi. Data yang diambil dari web service berupa data-data seperti data jadwal perkuliahan program studi, jadwal dosen mengajar, jadwal ruangan, daftar perwalian mahasiswa dan catatan perwaliannya.

3.2. Implementasi Program

Setelah selesai membuat *unit testing* maka selanjutnya membuat implementasi program yang akan di buat, hasil implementasi program dapat dilihat pada Gambar 3 dan Gambar 4. Pada Gambar 3 merupakan implementasi dari fitur simeru, ada 4 menu utama pada fitur simeru yaitu jadwal program studi digunakan untuk melihat jadwal perkuliahan, jadwal dosen untuk melihat jadwal dosen, jadwal ruang untuk melihat jadwal ruang yang digunakan selama masa perkuliahan, dan pengumuman digunakan untuk mengetahui informasi perkuliahan yang

diambil. Pada Gambar 4 merupakan implementasi dari fitur perwalian, ada 3 menu utama pada fitur perwalian yaitu transkrip nilai untuk melihat transkrip nilai dari mahasiswa, catatan perwalian digunakan untuk melakukan kegiatan perwalian kepada dosen wali, dan pengumuman dosen wali digunakan untuk melihat informasi dari dosen wali.



Gambar 3. Implementasi program simeru



Gambar 4. Implementasi program perwalian

3.3. Hasil Testing

Tahapan selanjutnya adalah melakukan testing dari hasil implementasi program, hasil testing dapat dilihat pada Gambar 5 sampai dengan Gambar 11. Pada Gambar 5, baris 2-11 berfungsi untuk mengambil data dari *webservice*, dan pada baris 12-13 berfungsi untuk membandingkan data dari *webservice*.

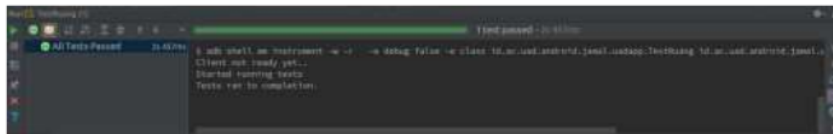
1.	@Test
2.	public void useApplicationContext() throws Exception{
3.	RuangCallBack personUad = new RuangCallBack(context);
4.	final SettableFuture<String> ff = SettableFuture.create();
5.	personUad.RuangCallBack("", new RuangCallBack.Callback() {
6.	@Override
7.	public void Result(String result) {
8.	Log.e("data" , result.toString());
9.	ff.set(result);
10.	}
11.	});
12.	String data = ff.get(5, TimeUnit.SECONDS);
13.	assertEquals(data, databenr);
14.	}

Gambar 5. Skema *Testing* Lihat Jadwal Ruang

Gambar 6 merupakan hasil *running* dari skema *testing*, hasil *running failed* karena belum ada data yang akan dites. Sedangkan Gambar 7 untuk menguji fitur jadwal ruangan dengan mengambil data berdasarkan sesuai program studi ketika user login dan hasil testing menunjukkan tidak ada yang *error* yang berarti bahwa fitur tersebut berjalan dengan baik.



Gambar 6. Hasil testing gagal pada fitur jadwal ruangan



Gambar 7. Hasil testing berhasil pada fitur jadwal ruangan

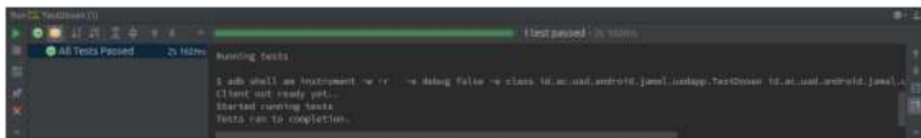
Pada Gambar 8 baris 2-10 berfungsi untuk mengambil data dari *web service*, dan pada baris 11-12 berfungsi untuk membandingkan data dari *web service*. Pada gambar 9 merupakan hasil running dari skema *testing*, hasil *running failed* karena belum ada data yang akan dites. Gambar 10 untuk menguji fitur jadwal dosen pada aplikasi dengan mengambil data dosen berdasarkan program studi ketika user login, hasil testing menunjukkan tidak ada *error* yang berarti fitur tersebut berjalan dengan baik.

1.	@Test
2.	public void useTestDosen() throws Exception{
3.	DosenCallBack dosen = new DosenCallBack(context);
4.	final SettableFuture<String> ff = SettableFuture.create();
5.	dosen.DosenCallBack("", new DosenCallBack.DsnCallBack()) {
6.	@Override
7.	public void Result(String result) {
8.	ff.set(result);
9.	}
10.	};
11.	String isi = ff.get(5, TimeUnit.SECONDS);
12.	assertEquals(isi, datanya);
13.	}

Gambar 8. Skema *Testing* Lihat Jadwal Dosen



Gambar 9. Hasil testing gagal pada fitur jadwal dosen



Gambar 10. Hasil testing berhasil pada fitur jadwal dosen

Pada Gambar 11 baris 2-8 berfungsi untuk mengambil data dari *web service*, dan pada baris 11-12 berfungsi untuk membandingkan data dari *webservice*. Pada Gambar 12 merupakan hasil *running* dari skema *testing*, hasil *running failed* karena belum ada data yang akan dites. Gambar 13 untuk menguji fitur jadwal kuliah pada aplikasi dengan mengambil data

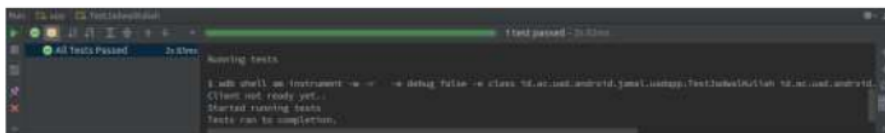
berdasarkan program studi *user login*, hasil testing menunjukkan tidak ada *error* yang berarti fitur berjalan dengan baik.

1.	@Test
2.	public void useTestJadwalKuliah() throws Exception{
3.	KuliahCallBack kuliah = new KuliahCallBack(context);
4.	final SettableFuture<String> ff = SettableFuture.create();
5.	kuliah.KuliahCallBack("", new KuliahCallBack.Callback() {
6.	@Override
7.	public void Result(String result) {
8.	ff.set(result);
9.	}
10.	});
11.	String isi = ff.get(5, TimeUnit.SECONDS);
12.	assertEquals(isi, Datanya);
13.	}

Gambar 11. Skema *Testing* Lihat Jadwal kuliah



Gambar 12. Hasil testing gagal pada fitur jadwal kuliah



Gambar 10. Hasil testing berhasil pada fitur jadwal kuliah

Pengujian lainnya juga telah dilakukan untuk fitur pengumuman simeru, transkrip nilai, catatan perwalian, pengumuman dosen wali, meski tidak disampaikan dalam artikel ini. Karena sifatnya iteratif seperti pada fitur jadwal-jadwal tersebut.

3.4. Pengujian *Usability*

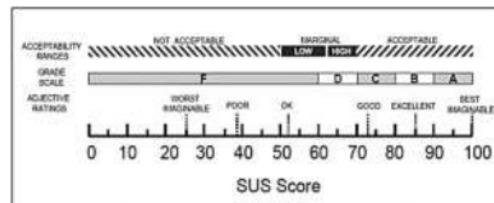
Tahapan selanjutnya melakukan uji *usability* untuk mengetahui seberapa mudah user dalam pengoprasian aplikasi yang telah dibuat uji *usability* menggunakan metode *System Usability Scale* (SUS) [23][24]. Berikut adalah 10 pertanyaan yang digunakan untuk melakukan uji *usability*:

1. Saya berpikir akan menggunakan sistem ini lagi?
2. Saya merasa sistem ini rumit untuk digunakan?
3. Saya merasa sistem ini mudah untuk digunakan?
4. Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini?
5. Saya merasa fitur-fitur sistem ini berjalan dengan semestinya?
6. Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini?
7. Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat?
8. Saya merasa sistem ini membingungkan?
9. Saya merasa tidak ada hambatan dalam menggunakan sistem ini?
10. Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini?

Hasil pengujian *usability* yang diperoleh dari 9 responden yang telah menjawab pertanyaan (Q1-Q10) ditampilkan pada rekapitulasi total skor pada Tabel 1.

Tabel 1. Rekapitulasi total skor

Responden	Total skor pertanyaan										Total
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	
1	3	3	3	4	3	3	3	4	3	3	80
2	3	2	3	2	3	1	3	2	3	3	62,5
3	4	2	3	2	2	2	2	2	3	2	60
4	3	2	3	1	3	1	4	3	3	2	62,5
5	4	3	3	4	3	4	4	3	4	3	87,5
6	3	3	3	2	3	3	3	3	3	3	72,5
7	3	3	3	3	3	3	4	3	3	3	77,5
8	3	2	3	2	3	3	1	3	3	3	65
9	3	2	3	2	3	2	3	1	2	3	60



Gambar 12. Peringkat skor SUS [23].

Berdasarkan hasil pengolahan data responden pada Tabel 1 maka didapat nilai akhir SUS dapat dihitung sebagai berikut berdasarkan kolom skor total dari tiap pertanyaan, yaitu:

$$(80 + 62,5 + 60 + 62,5 + 87,5 + 72,5 + 77,5 + 65 + 60)/9 = 69,7 \quad (1)$$

Gambar 12 merupakan referensi skor SUS yang akan digunakan untuk kalibrasi hasil perhitungan skor SUS. Pada Gambar 12 terlihat terdapat tiga kategori yaitu terkait dengan *Acceptability Ranges*, *Grade Scale* dan *Adjective Ratings*. Dengan skor hasil perhitungan SUS penelitian ini yaitu 69,7 dapat diartikan bahwa aplikasi ini dinyatakan bisa diterima (*acceptable*) pada kategori *Acceptability Ranges*. Sedangkan pada katagori *Grade Scale* penilaian aplikasi mendapatkan grade D, dan sedangkan pada penilaian *Adjective Ratings* aplikasi masuk pada katogori "Oke". Sehingga dapat dikatakan bahwa aplikasi meski telah sukses melewati semua *unit testing*, namun aplikasi ini masih dapat dikembangkan lagi dari sisi *user interface*-nya untuk meningkatkan nilai *usability* dari penggunaanya.

4. Kesimpulan

Berdasarkan hasil pembahasan, maka telah dikembangkan aplikasi *android* perwalian dan simeru dengan pendekatan *Test Driven Development* yang dapat digunakan mahasiswa Universitas Ahmad Dahlan untuk melakukan kegiatan perwalian, mengecek informasi perkuliahan, jadwal dosen, jadwal program studi dan jadwal ruang. Hasil pengujian usability yaitu pengujian menggunakan *System Usability Scale* (SUS) yang menghasilkan nilai 69,7 menunjukan system dapat diterima.

Referensi

- [1]. Setiawan, A., Setiaji, P., & Utomo, A. P. (2011). Sistem Informasi Perwalian pada Program Studi Sistem Informasi Universitas Muria Kudus. *Sains Dan Teknologi*, 4(1), 1–18.
- [2]. Setyadi, H. A., & Nugroho, E. C. (2014). Pengembangan Sistem Bagi Pembimbing Akademik Untuk Memantau Perkembangan Mahasiswa. *Sentra Penelitian Engineering Dan Edukasi*, 6(3), 57–65.
- [3]. Kartika, F., Dewi, S., Indriasari, T. D., & Prayogo, Y. (2016). Rancang Bangun Aplikasi Pengingat Kegiatan Akademik Berbasis Mobile. *Jurnal Buana Informatika*, 7(4), 303–312.
- [4]. Laksito, A. D. (2013). Sinkronisasi Jadwal Perkuliahan pada Aplikasi Android menggunakan Teknologi XML-RPC (Studi Kasus di STMIK AMIKOM Yogyakarta). In *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)* (pp. 1–5).

- [5]. Prasetyo, A. D., Rahmad, C., Informasi, T., & Malang, P. N. (2015). PERANCANGAN APLIKASI SCHEDULE REMINDERS MENGGUNAKAN METODE EUCLIDEAN DISTANCE. In *Prosiding Seminar Informatika Aplikatif Polinema 2015* (pp. 126–128).
- [6]. Ramadhan, T., & Utomo, V. (2014). Rancang Bangun Aplikasi Mobile Untuk Notifikasi Jadwal Kuliah Berbasis Android (Studi Kasus: Stmik Provisi Semarang). *Jurnal Teknologi Informasi Dan Komunikasi*, 5(2), 47–55.
- [7]. Silvia, L., Dengen, N., & Hairah, U. (2017). Fakultas Ilmu Komputer Dan Teknologi Informasi Berbasis Android (Studi Kasus : Fakultas Ilmu Komputer Dan Teknologi Informasi). In *Prosiding Seminar Nasional Ilmu Komputer dan Teknologi Informasi* (Vol. 2, pp. 2540–2542).
- [8]. Sukmandari, D. G., & Sukardiyono, T. (2017). Analisis aplikasi jadwal perkuliahan berbasis android. *Elinvo (Electronics, Informatics, and Vocational Education)*, 2(1), 80–84.
- [9]. Yuliansyah, H, Wibowo, M., Arisandy, W., Budiarti, A. "Rancangan Aplikasi Informasi Jadwal Perkuliahan Berbasis Mobile (Studi Kasus Fakultas Teknologi Industri UAD)". *Seminar Teknik Informatika 2013*. 27 Juni 2013 .
- [10]. Abdurahman, M. (2016). Sistem Informasi Jadwal Perkuliahan Berbasis Web Mobile Pada Politeknik Sains Dan Teknologi Wiratama Maluku Utara Muhdar Abdurahman Politeknik Sains Dan Teknologi Wiratama Maluku Utara. *Indonesian Journal on Networking and Security*, 5(2), 49–56.
- [11]. Handaka, S. H., & Kuntadi, E. (2015). Pengembangan Sistem Perwalian Akademik Berbasis WEB (Studi Kasus Pada Jurusan Manajemen Fakultas Ekonomi Universitas Negeri Semarang). *Jurnal Teknologi Informasi Dan Komunikasi*, 6, 25–30.
- [12]. Sanjaya, ginanjar tegar, & Sumboro, B. (2015). Rancang Bangun Aplikasi Penjadwalan Kuliah STMIK AUB Surakarta Berbasis Web. *Jurnal Go Infotech*, 21(1), 37–42.
- [13]. Usada, E., Yuniarsyah, Y., & Rifani, N. (2012). Rancang Bangun Sistem Informasi Jadwal Perkuliahan Berbasis JQuery Mobile Dengan Menggunakan PHP dan MySQL. *Jurnal Infotel*, 4(2), 40–51.
- [14]. Irawan, josep dedy, & Adriantantri, E. (2018). Pembuatan Aplikasi Penjadwalan Kuliah. *Jurnal MNEMONIC*, 1(1), 70–74.
- [15]. Hanief, S. (2016). Sistem Perwalian Online Mahasiswa Pada Program Pendidikan Jarak Jauh. *Jurnal Sistem Dan Informatika*, 10, 86–92.
- [16]. Nyoman, N., & Januhari, U. (2015). Perancangan Sistem Informasi Peminjaman Penggunaan Ruangan pada STMIK STIKOM Bali. *Jurnal Sistem Dan Informatika*, 9(2), 86–94.
- [17]. Pandhita, A. L., Yuliansyah, H. (2016). Rancang Bangun Aplikasi Android Berbasis Lokasi Untuk Pencarian Distribution Point. *Jurnal Infotel*, 7(2), 83-92.
- [18]. Wibowo, M., Yuliansyah, H. "Location Based Service Application Design for Mobile Promotion SME's Product and the Nearest Bank Service Information". *International Conference on Green World in Business and Technology 2014*. 29 Maret 2014 2014.
- [19]. Rais, A., Yuliansyah, H. (2015). Apikasi Pemandu Museum Gunungapi Merapi (Mgm) Dengan Konsep Layanan Berbasis Lokasi Dalam Ruangan Menggunakan Qr Code. *Jurnal Informatika*, 9(2), 1095-1105.
- [20]. Yuliansyah, H., Novitasari I. D. (2015). M-Zoo: Aplikasi Android Dengan Konsep Layanan Berbasis Lokasi Untuk Informasi Pengenalan Satwa di Kebun Binatang. *TECHSI-Jurnal Teknik Informatika*, 7(2), 171-183.
- [21]. Pamungkas, G., Yuliansyah, H. (2017). Rancang Bangun Aplikasi Android Pos (Point of Sale) Kafe Untuk Kasir Portable dan Bluetooth Printer. *JST (Jurnal Sains dan Teknologi)*, 6(1), 199-208.
- [22]. Khan, B. N., Khan, M. S., Javed, M. A., & Khan, M. A. (2013). Reducing Testing Effort in the Test Driven Development. *Global Journal of Computer Science and Technology*, 13(7), 0–4.
- [23]. Brooke, J. (2013). SUS : A Retrospective. *Journal of Usability Studies*, 8(2), 29–40. <https://doi.org/10.1074/jbc.R115.675280>
- [24]. Sus, U. S., Santoso, H. B., Sharfina, Z., & Santoso, H. B. (2016). An Indonesian Adaptation of the System An Indonesian Adaptation of the System Usability Scale (SUS).

HASIL CEK_Implementasi Test Driven

ORIGINALITY REPORT

8%

SIMILARITY INDEX

9%

INTERNET SOURCES

9%

PUBLICATIONS

8%

STUDENT PAPERS

PRIMARY SOURCES

1

jurusan.tik.pnj.ac.id

Internet Source

4%

2

Submitted to Syiah Kuala University

Student Paper

4%

Exclude quotes On

Exclude matches < 4%

Exclude bibliography On