

Pemanfaatan Restful Api Pada Mobile based test Untuk Sertifikasi Karyawan

Anggraini Kusumaningrum
Program studi Informatika
Institut Teknologi Dirgantara
Adisutjipto
Yogyakarta, Indonesia
anggraini@itda.ac.id

Yuliani Indrianingsih
Program studi Informatika
Institut Teknologi Dirgantara
Adisutjipto
Yogyakarta, Indonesia
yuli.stta@gmail.com

Asih Pujiastuti
Program studi Informatika
Institut Teknologi Dirgantara
Adisutjipto
Yogyakarta, Indonesia
asih@itda.ac.id

Mardiana Irawaty
Program studi Informatika
Institut Teknologi Dirgantara
Adisutjipto
Yogyakarta, Indonesia
mi80006@yahoo.com

Satya Wira Wicaksana
Program studi Teknik Informatika
Sekolah Tinggi Teknologi Adisutjipto
Yogyakarta, Indonesia
satya.wiraw@gmail.com

Abstract— A test is something that is used to test the quality of intelligence, ability, learning outcomes and so on. The test is always associated with an assessment or evaluation of a person to determine the person's mastery of the material. The rapid development of the digital world makes it easier for people to complete their work, the use of mobile tests is one of the technologies used by using the Application Programming Interface (API). Restful API is one of the architectures in the API which has 4 important components, namely URL Design, HTTP Verbs, HTTP Response Code, and Response Format. Rest client will access data to the Rest server, each data is distinguished based on Global ID or Universal Resource Identifier (URI) in the form of formal text XML or JSON. From the test results using WhiteBox on the admin side, there are 4 paths for 3 days with 100% success. While on the client side there are 8 paths for 3 days with 100% success. Testing Time synchronization of data with 10 to 30 users for 5 times of testing obtained 10.8 seconds

Keywords— *Restful API, Mobile Test, WhiteBox*

Abstrak— Test merupakan sesuatu yang dipakai untuk menguji mutu kepandaian, kemampuan, hasil belajar dan sebagainya. Test selalu dihubungkan dengan penilaian atau bahan evaluasi terhadap seseorang untuk menentukan seseorang tersebut terhadap penguasaan suatu materi. Perkembangan dunia digital yang pesat memudahkan orang untuk menyelesaikan pekerjaannya, penggunaan mobile test merupakan salah satu teknologi yang digunakan dengan menggunakan *Application Programming Interface (API)*. Restful API merupakan salah satu arsitektur yang ada di API yang memiliki 4 komponen penting yaitu URL Design, HTTP Verbs, HTTP Response Code, Format response. Rest client akan melakukan akses data ke Rest server, masing-masing data dibedakan berdasarkan Global ID atau Universal Resource Identifier (URI) berupa formal text XML atau JSON. Dari hasil pengujian dengan menggunakan WhiteBox disisi admin terdapat 4 path selama 3 hari dengan keberhasilan 100%. Sedangkan di sisi client terdapat 8 path selama 3 hari dengan keberhasilan 100%. Pengujian Waktu sinkronisasi data dengan 10 sampai 30 pengguna untuk 5 kali pengujian diperoleh waktu 10,8 detik.

Keywords— *Restful API, Mobile Test, WhiteBox*

PENDAHULUAN

Menurut kamus besar Bahasa Indonesia, test merupakan sesuatu yang dipakai untuk menguji mutu kepandaian, kemampuan, hasil belajar dan sebagainya. Tes selalu dihubungkan dengan penilaian atau bahan evaluasi terhadap seseorang untuk menentukan seseorang tentang penguasaan suatu materi. Hasil dari tes salah satu nya digunakan untuk membuat keputusan lulus atau tidaknya atas materi tersebut. Hasil tes di anggap sebagai bukti yang valid dari individu yang dapat digunakan, misalnya untuk kenaikan kelas, promosi jabatan, dan kelulusan. Saat ini, Go Green telah menjadi inisiatif dalam melindungi sumber daya alam untuk generasi berikutnya dan melindungi kesehatan manusia melalui pengelolaan lingkungan dan penerapan cara hidup hijau. proses pelaksanaan Tes berbasis kertas untuk peningkatan Sumber Daya Manusia (SDM) karyawan maupun pengambilan sertifikasi dirasa kurang efektif dan efisien dari segi waktu. Dikarenakan akan memakan waktu yang lama untuk mengoreksi hasil tes satu persatu.

Penggunaan moodleble pada mobile learning dikembangkan secara khusus dan dioptimasi sedemikian rupa untuk dapat kompatibel pada perangkat dan sumberdaya yang terbatas serta memiliki ineteroperabilitas yang tinggi[1]. Rancang Bangun Aplikasi Elektronik Transportasi Bandara Yogyakarta International Airport Berbasis Mobile Menggunakan Android Studio[2]. Pengembangan Sistem Stock Opname Berbasis Mobile Application Using SDLC Methode mempermudah petugas dalam proses stock opname karena sistem mobile apps cukup menggunakan satu perangkat cerdas yang mempermudah user dalam menginput data stock dengan cepat sehingga meningkatkan produktivitas, kecepatan dan keefektifan[3]. Implementasi Algoritma Random pada Computer base Test untuk penerimaan mahasiswa baru STTA yogyakarta[4]. Perancangan Aplikasi Ujian Saringan Masuk Perguruan Tinggi Secara Online Berbasis Android Studi Kasus Universitas Potensi Utama Medan menyatakan bahwa dalam perkembangan teknologi informasi telah menyentuh hampir semua sektor[5]. evaluasi pada suatu kegiatan pembelajaran sangat lah penting sebagai tolak ukur pemahaman bagi penerima materi pembelajaran dengan Implementasi Computer Based Test (CBT) di Sekolah Menengah Kejuruan[6]. Rancang Bangun Aplikasi Ujian Online Pra Kompre Berbasis Android, untuk membantu dosen dalam melaksanakan ujian pra kompre kepada mahasiswa nya[7]. Perbandingan Rest Api dan SOAP pada teknologi web service

untuk android[8]. Implementasi RestAPI web Service dalam membangun aplikasi multiplatform untuk usaha Jasa menggunakan metode waterfall dalam penyelesaian permasalahan, hasil pengujian responden tukang memperoleh hasil 51,4% sangat setuju[9]. Penggunaan RestAPI dalam rancang bangun Weshare sebagai upaya mempermudah pelayanan donasi kemanusiaan memberikan kemudahan transaksi donasi[10]. RestAPI Web service menggunakan JSON Web Token untuk system pencatatan transaksi[11] Berbeda dengan penelitian sebelumnya tentang ujian online belum memanfaatkan RestAPI sebagai web Service dan pada aplikasi-aplikasi yang memanfaatkan RestAPI belum sepenuhnya dimanfaatkan untuk sertifikasi karyawan yang bertujuan untuk membuat aplikasi test berbasis Android dengan memanfaatkan RestAPI sehingga waktu yang dibutuhkan lebih cepat dibandingkan dengan menggunakan Web Service SOAP.

Kemudahan implementasi	Mudah
------------------------	-------

METODOLOGI PENELITIAN

A. Sertifikasi

Pengertian sertifikasi adalah pemberian sertifikat melalui suatu pengujian yang didasarkan pada standar jabatan atau persyaratan pekerjaan yang berlaku secara nasional/standar kompetensi[12].

B. Restful API

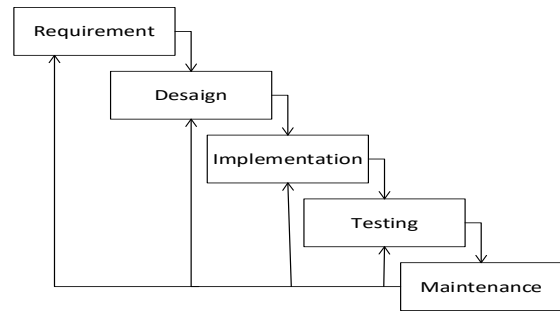
API memungkinkan developer untuk mengintegrasikan dua bagian dari aplikasi atau dengan aplikasi yang berbeda secara bersamaan. API terdiri dari berbagai elemen seperti function, protocols, dan tools lainnya yang memungkinkan developers untuk membuat aplikasi. Teknologi Web service dapat diimplementasikan dengan ReST (Representational State Transfer), Tabel 1 merupakan penjelasan tentang teknologi Restful API.

TABEL I. TEKNOLOGI RESTFUL API

Uraian	ReSTful
Protokol komunikasi	HTTP, HTTPS
Penggunaan bandwidth	Relatif hemat bandwidth, karena markup-markup ekstra seperti pada XML tidak dipakai
Tren penggunaan	Mulai populer, banyak dipakai oleh penyedia web servis terkemuka, seperti twitter, yahoo!, flickr, bloglines, technorati, google, amazon, eBay, dan sebagainya
Aturan penulisan	Tidak ada spesifikasi khusus
Format respon	XML, JSON, atau format plain teks lainnya. Hal ini memudahkan penerima respon membaca dan memahaminya.
Attachment file	Tidak Bisa
Sifat web servis pada umumnya	Terbuka, bisa diakses siapa saja (web API)
Caching web	Mudah, karena menggunakan URI
Penggunaan standar	Standar yang sudah ada, seperti XML dan HTTP
Tool pengembangan	Beberapa, karena tidak begitu dibutuhkan
Tool manajemen	Menggunakan tool yang sudah ada pada sistem jaringan
Ekstensibel	Relatif tidak ekstensibel

C. LANGKAH-LANGKAH PENYELESAIAN

Penelitian dilakukan di Garuda Maintenance Facility (GMF) dengan menggunakan Water fall, Metode waterfall menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak yang dimulai dari requirement analysis, system and software design, implementation, testing and maintenance[13]. Gambar I merupakan model water fall.



GAMBAR I. METODE WATER FALL

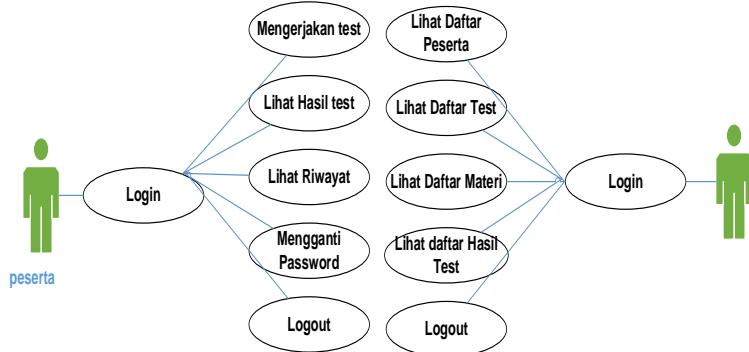
1) Tahapan Requirement

Pada proses pembuatan aplikasi membutuhkan data-data yang berkaitan dalam proses sertifikasi karyawan PT. GMF AeroAsia. Data-data tersebut meliputi: Data keryawan dan Data Bank soal.

2) Tahapan Design

a) Use Case Diagram

Use case digunakan mendiskripsikan sebuah interaksi antar satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi. Pada aplikasi mobile based test untuk sertifikasi karyawan Peserta memiliki 6 case yaitu login, mengerjakan tes, melihat hasil tes, melihat riwayat tes, mengganti password, dan melakukan logout. Sedangkan admin memiliki 6 case juga yaitu login, melihat daftar peserta, melihat daftar tes, melihat daftar materi, melihat daftar hasil test, dan logout. Gambar II. Merupakan Use Case aplikasi.

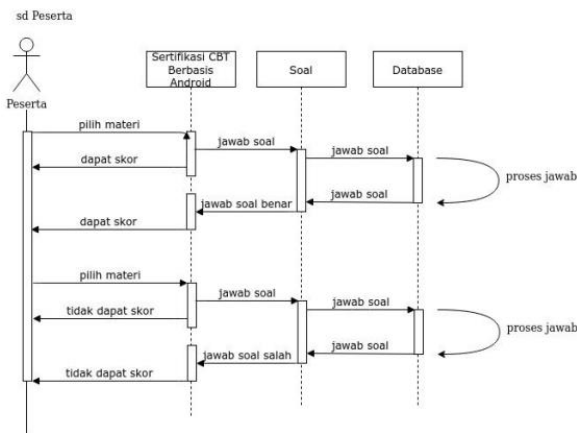


GAMBAR II. USE CASE DIAGRAM

b) Sequence Diagram

Sequence diagram menjelaskan tentang interaksi objek yang disusun berdasarkan urutan waktu. Peserta akan memilih

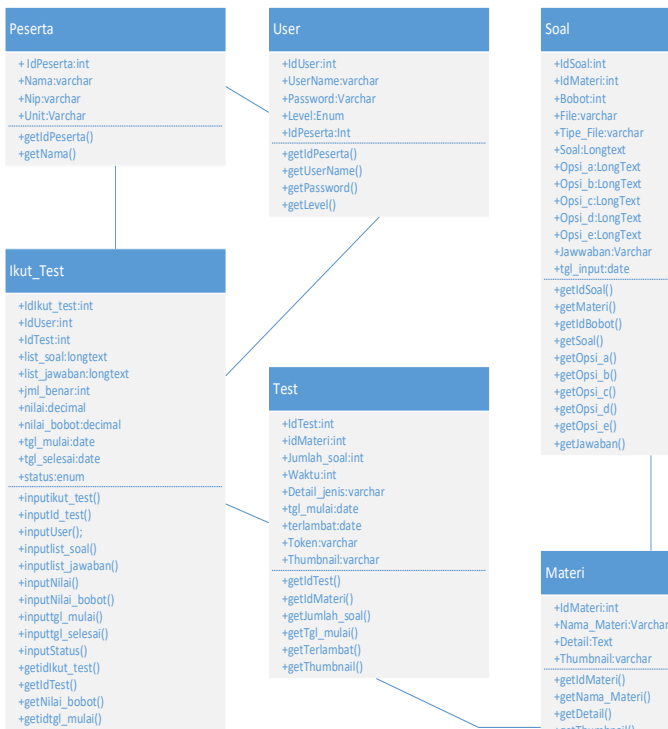
materi yang akan diujikan untuk sertifikasi karyawan, kemudian peserta akan menjawab soal-soal yang telah ditentukan oleh admin. Jika jawaban benar maka akan mendapatkan skor, namun jika soal salah maka tidak akan mendapatkan skor. Gambar III merupakan gambar sequence diagram untuk proses pelaksanaan test sertifikasi karyawan.



GAMBAR III. SEQUENCE DIAGRAM APLIKASI

c) Class Diagram

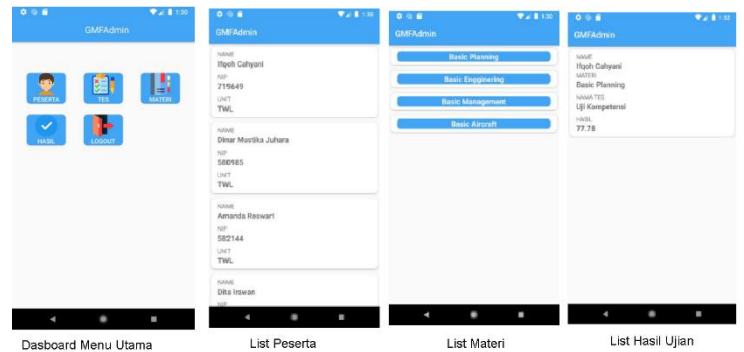
Class Diagram menggambarkan atribut, property, atau metode pada suatu abstract class beserta hubungan antar class yang lain. pada Class diagram aplikasi mobile based test sertifikasi karyawan menunjukkan bahwa sistem tersebut menggunakan 6 class yang mewakili masing-masing objek yang ada, yaitu Peserta, Soal, User, Materi, Tes, dan Ikut_tes. Class Peserta bergantung pada class User begitu juga dengan class Soal yang bergantung pada class Materi. Pada class Tes bergantung dengan class Materi dan juga class Ikut_tes bergantung pada class User dan Materi. Gambar IV merupakan class diagram aplikasi.



GAMBAR IV. CLASS DIAGRAM APLIKASI

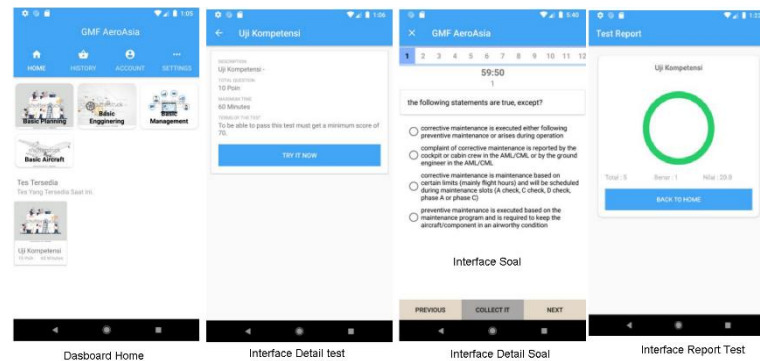
3) Implementation

Halaman Admin, di khususkan bagi Administrator, Di PT. GMF AeroAsia dilakukan oleh divisi learning Center. Selain login, kegiatan yang dilakukan oleh Admin adalah melihat dasboar menu utama, melihat list peserta, list Materi dan list hasil. Gambar V merupakan implementasi di sisi Admin.



GAMBAR V. IMPLEMENTASI HALAMAN ADMIN

Halaman User di khususkan untuk karyawan yang akan melakukan Test Kompetensi. Implementasi pada halaman user meliputi, Login, Dashboard Home, memilih kategori, detail test, soal, report hasil dan history test. Gambar VI merupakan implementasi di sisi user.



GAMBAR VI. IMPLEMENTASI HALAMAN USER

4) Testing

Proses testing digunakan untuk menguji aplikasi, baik dari segi fungsionalitas dan performa aplikasi .

5) Maintenance

Tahapan maintenance ini dilakukan untuk memperbaiki system jika terdapat kegagalan proses yang tidak terdeteksi saat proses Testing.

HASIL DAN PEMBAHASAN

Pengujian aplikasi bertujuan untuk investigasi terhadap kualitas dari suatu produk. Dalam tugas akhir ini terdapat dua pengujian yaitu uji fungsionalitas dan uji.

A. UJI FUNGSIONALITAS

Pengujian fungsionalitas aplikasi dilakukan oleh unit Learning Service selaku unit yang akan menggunakan aplikasi ini sebagai opsi untuk melaksanakan tes sertifikasi pada karyawan. Dari hasil uji fungsionalitas 100% aplikasi berhasil di jalankan sesuai dengan perancangan. Tabel II merupakan

hasil uji fungsionalitas admin. Sedangkan Tabel III merupakan uji fungsionalitas di sisi peserta.

TABEL II. UJI FUNGSIONALITAS ADMIN

Fungsi	Berhasil	Gagal
Button Login	✓	
Button Peserta	✓	
Button Test	✓	
Button Materi	✓	
Button Hasil	✓	
Button Signout	✓	

TABEL III. UJI FUNGSIONALITAS PESERTA

Fungsi	Berhasil	Gagal
Button Login	✓	
Button Dashboard Home	✓	
Button Dashboard History	✓	
Button Dashboard Account	✓	
Button Dashboard Setting	✓	
Button Cardview Materi	✓	
Button Cardview Test	✓	
Button Start Test	✓	
Radio Button Answer	✓	
Button Previous	✓	
Button Collect It	✓	
Button Next	✓	
Button Report	✓	
Button Change Password	✓	
Button Change	✓	
Button About Us	✓	
Button SignOut	✓	

B. UJI WHITEBOX

Pengujian whitebox merupakan pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program. Pengujian Whitebox digunakan untuk meyakinkan semua perintah dan kondisi pada aplikasi dieksekusi secara minimal. Pada pengujian Whitebox menggunakan flowgraph untuk menggambarkan alur dari proses.

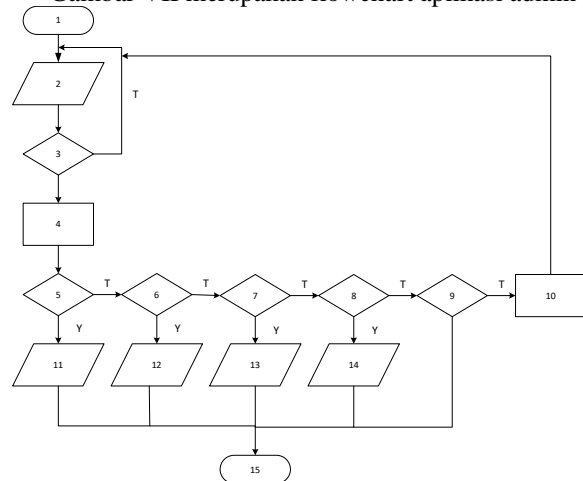
1. APLIKASI DI SISI ADMIN

Langkah 1. Membuat flowchart

- 1) Simbol 1 menunjukan proses mulai dari flowchart,
- 2) Simbol 2 menunjukan tampilan screen login,
- 3) Simbol 3 merupakan percabangan dari benar atau salah dari data yang dimasukkan saat login,
- 4) Simbol 4 menunjukkan tampilan screen utama aplikasi,
- 5) Simbol 5 menunjukkan percabangan apakah ingin menuju tampilan screen daftar peserta atau tidak,

- 6) Simbol 6 menunjukan percabangan apakah ingin menuju tampilan screen daftar tes atau tidak,
- 7) Simbol 7 menunjukkan percabangan apakah ingin menuju tampilan screen daftar materi atau tidak,
- 8) Simbol 8 menunjukkan percabangan apakah ingin menuju tampilan screen daftar tes atau tidak,
- 9) simbol 9 menunjukan percabangan apakah ingin logout dan menuju screen login atau tidak,
- 10) Simbol 10 menunjukan proses logout dari aplikasi,
- 11) Simbol 11 menunjukkan tampilan screen daftar peserta,
- 12) Simbol 12 menunjukkan tampilan screen daftar tes,
- 13) Simbol 13 menunjukkan tampilan screen daftar materi,
- 14) Simbol 14 menunjukkan tampilan screen daftar hasil,
- 15) Simbol 15 menunjukkan proses selesai dari flowchart.

Gambar VII merupakan flowchart aplikasi admin



GAMBAR VII. FLOWCHART ADMIN

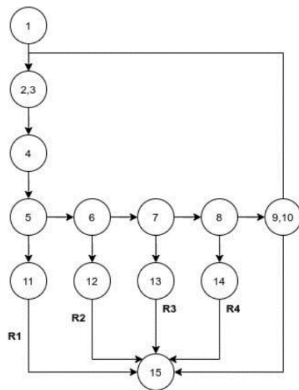
Langkah 2. Membuat flowgraph

- 1) Kondisi 1 menunjukan kondisi mulai.
- 2) Kondisi 2,3 menunjukan kondisi login dan percabangan.
- 3) Kondisi 4 menunjukan kondisi screen utama aplikasi.
- 4) Kondisi 5 menunjukan kondisi percabangan menuju screen daftar peserta atau tidak.
- 5) Kondisi 6 menunjukan kondisi percabangan menuju screen daftar tes atau tidak.
- 6) Kondisi 7 menunjukan kondisi percabangan menuju screen daftar materi atau tidak.
- 7) Kondisi 8 menunjukan kondisi percabangan menuju screen daftar hasil atau tidak
- 8) Kondisi 9,10 menunjukan kondisi percabangan dan jika iya maka akan terjadi proses logout dan menuju halaman login, jika tidak maka proses selesai.
- 9) Kondisi 11 menunjukan kondisi screen daftar peserta.
- 10)Kondisi 12 menunjukan kondisi screen daftar tes.
- 11)Kondisi 13 menunjukan kondisi screen daftar materi.

12) Kondisi 14 menunjukan kondisi screen daftar hasil.

13) Kondisi 15 menunjukan kondisi selesai.

Gambar VIII merupakan flowgraph aplikasi admin



GAMBAR VIII. FLOWGRAPH ADMIN

Langkah 3. Membuat Cyclomatic Complexity

Untuk mengetahui jumlah path yang perlu dilaksanakan agar semua proses dilewati. Tabel IV merupakan tabel cyclomatic complexity dengan tahapan alur sebagai berikut :

- Path 1 Mulai – login – percabangan – screen menu utama – percabangan – daftar peserta – selesai.
- Path 2 Mulai – login – percabangan – screen menu utama – percabangan – percabangan – daftar tes – selesai
- Path 3 Mulai – login – percabangan – screen menu utama – percabangan – percabangan – percabangan – daftar materi – selesai.
- Path 4 Mulai – login – percabangan – screen menu utama – percabangan – percabangan – percabangan – percabangan – daftar hasil – selesai

TABEL IV. CYLOMATIC COMPLEXITY ADMIN

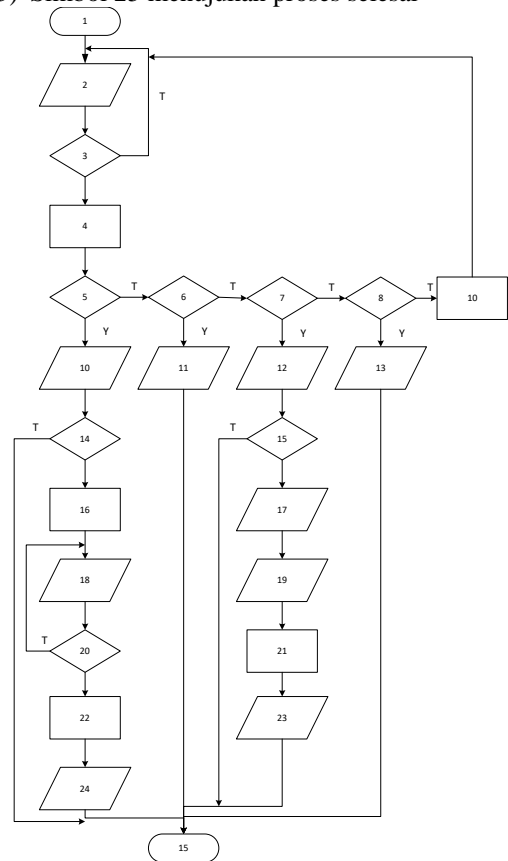
Path 1	1-2-3-4-5-11-15
Path 2	1-2-3-4-5-6-12-15
Path 3	1-2-3-4-5-6-7-13-15
Path 4	1-2-3-4-5-6-7-8-14-15

2. APLIKASI DI SISI PESERTA

Langkah 1. Membuat Flowchart

- 1) Simbol 1 menunjukkan proses mulai dari flowchart.
- 2) Simbol 2 menunjukan tampilan screen login.
- 3) Simbol 3 merupakan percabangan dari benar atau salah dari data yang dimasukkan saat login.
- 4) Simbol 4 menunjukkan tampilan screen dashboard menu utama.
- 5) Simbol 5 menunjukkan percabangan apakah ingin menuju tampilan screen info materi atau tidak.
- 6) Simbol 6 menunjukkan percabangan apakah ingin menuju tampilan screen daftar tes atau tidak.
- 7) Simbol 7 menunjukan percabangan apakah ingin menuju tampilan screen info akun atau tidak.
- 8) Simbol 8 menunjukan percabangan apakah ingin menuju tampilan screen about us atau tidak.
- 9) Simbol 9 menunjukkan proses logout.

- 10) Simbol 10 menunjukan tampilan screen info materi.
- 11) Simbol 11 menunjukan tampilan screen daftar riwayat tes.
- 12) Simbol 12 menunjukan tampilan info akun.
- 13) Simbol 13 menunjukan tampilan info about us.
- 14) Simbol 14 menunjukan percabangan apakah ada tes atau tidak.
- 15) Simbol 15 menunjukkan percabangan apakah ganti password atau tidak.
- 16) Simbol 16 menunjukan proses memulai tes.
- 17) Simbol 17 menunjukkan tampilan screen memasukkan password lama.
- 18) Simbol 18 menunjukkan tampilan screen saat menjawab soal.
- 19) Simbol 19 menunjukan tampilan screen memasukkan password baru.
- 20) Simbol 20 menunjukan percabangan sudah selesai mengerjakan tes atau belum
- 21) Simbol 21 menunjukan proses menyimpan password baru.
- 22) Simbol 22 menunjukan proses menghitung hasil tes.
- 23) Simbol 23 menunjukan proses tampilan password tersimpan.
- 24) Simbol 24 menunjukan tampilan screen hasil nilai tes.
- 25) Simbol 25 menunjukan proses selesai



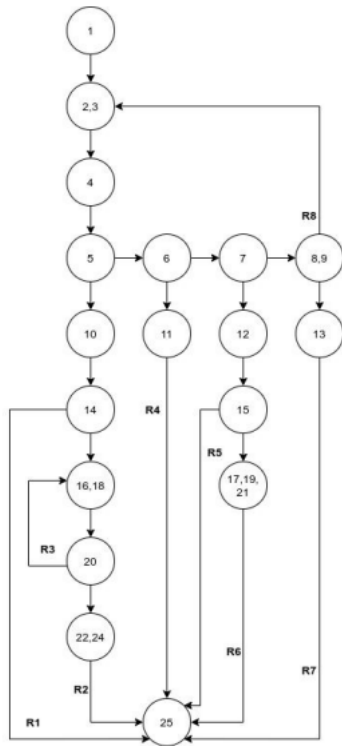
GAMBAR IX. FLOWCHART APLIKASI PESERTA

Langkah 2. Membuat Flowgraph

- 1) Kondisi 1 menunjukan kondisi mulai.
- 2) Kondisi 2,3 menunjukan kondisi login dan percabangan.
- 3) Kondisi 4 menunjukan kondisi screen utama aplikasi.
- 4) Kondisi 5 menunjukan percabangan menuju screen info materi atau tidak.

- 5) Kondisi 6 menunjukan percabangan menuju screen daftar riwayat atau tidak.
- 6) Kondisi 7 menunjukan percabangan menuju screen info akun atau tidak.
- 7) Kondisi 8 menunjukan percabangan menuju screen setelan atau tidak.
- 8) Kondisi 9 menunjukan screen info materi.
- 9) Kondisi 10 menunjukan screen info riwayat.
- 10) Kondisi 11 menunjukan proses info akun.
- 11) Kondisi 12 menunjukan screen about us.
- 12) Kondisi 13 menunjukan percabangan ada tes atau tidak.
- 13) Kondisi 14 menunjukan percabangan ganti password atau tidak.
- 14) Kondisi 15 menunjukan percabangan logout atau tidak.
15. Kondisi 16,19 menunjukan proses mulai tes.
- 15) Kondisi 17,20,22 menunjukan proses pergantian password.
- 16) Kondisi 18 menunjukan proses logout.
- 17) Kondisi 21 menunjukan percabangan apakah sudah selesai mengerjakan tes atau belum.
- 18) Kondisi 23,24 menunjukan proses perhitungan hasil tes dan tampilan screen hasil tes.
- 19) Kondisi 25 menunjukan proses selesai

- Path 3 Mulai – login – percabangan – menu utama – percabangan – info materi – percabangan – mulai tes – menjawab soal – percabangan – menjawab soal
- Path 4 Mulai – login – percabangan – menu utama – percabangan percabangan – tampil detail riwayat tes – selesai.
- Path 5 Mulai – login – percabangan – menu utama – percabangan percabangan – percabangan – tampil info akun – percabangan – selesai.
- Path 6 Mulai – login – percabangan – menu utama – percabangan percabangan – percabangan – info akun – percabangan – screen password lama – screen password baru – simpan password – password tersimpan – selesai.
- Path 7 Mulai – login – percabangan – menu utama – percabangan percabangan – percabangan – percabangan – tampil about us – selesai.
- Path 8 Mulai – login – percabangan – menu utama – percabangan percabangan – percabangan – percabangan – logout – login



GAMBAR X. FLOWGRAPH

Langkah 3. Membuat Cyclomatic Complexity

Untuk mengetahui jumlah path yang perlu dilaksanakan agar semua proses dilewati. Tabel V merupakan tabel cyclomatic complexity dengan tahapan alur sebagai berikut :

- Path 1 Mulai – login – percabangan – menu utama – percabangan – info materi – percabangan – selesai.
- Path 2 Mulai – login – percabangan – menu utama – percabangan – info materi – percabangan – mulai tes – menjawab soal – percabangan – penghitungan hasil tes – tampil detail nilai tes – selesai.

TABEL V. CYLOMATIC COMPLEXITY PESERTA

Path 1	1-2-3-4-5-10-14-25
Path 2	1-2-3-4-5-10-14-16-18-20-22-24-25
Path 3	1-2-3-4-5-10-14-16-18-20-16
Path 4	1-2-3-4-5-6-11-25
Path 5	1-2-3-4-5-6-7-12-15-25
Path 6	1-2-3-4-5-6-7-12-15-17-19-21-23-25
Path 7	1-2-3-4-5-6-7-8-9-13-25
Path 8	1-2-3-4-5-6-7-8-9-2

Dari hasil pengujian dengan menggunakan whitebox selama 3 hari, diperoleh hasil 100% sesuai dengan alur cyclomatic complexity. Tabel VI merupakan hasil pengujian whitebox untuk admin dan Tabel VII merupakan hasil pengujian whitebox untuk peserta.

TABEL VI. WHITEBOX ADMIN

Path cyclomatic complexity	Hari 1		Hari 2		Hari 3	
	Ya	Tidak	Ya	Tidak	Ya	Tidak
Path 1	✓		✓		✓	
Path 2	✓		✓		✓	
Path 3	✓		✓		✓	
Path 4	✓		✓		✓	

TABEL VII. WHITEBOX PESERTA

Path cyclomatic complexity	Hari 1		Hari 2		Hari 3	
	Ya	Tidak	Ya	Tidak	Ya	Tidak
Path 1	✓		✓		✓	
Path 2	✓		✓		✓	
Path 3	✓		✓		✓	
Path 4	✓		✓		✓	
Path 5	✓		✓		✓	
Path 6	✓		✓		✓	

Path 7	V		V		V	
Path 8	V		V		V	

C. PENGUJIAN WAKTU SINKRONISASI DATA

Pengujian dilakukan pada aplikasi mobile test yang memanfaatkan Restful API sebagai web service. Dari hasil pengujian waktu sinkronisasi diperoleh hasil penggunaan Restful API diperoleh rata-rata waktu 10,8 detik. Tabel VIII merupakan tabel yang menunjukkan waktu sinkronisasi data.

TABEL VIII. TABEL WAKTU SINKRONISASI DATA

Jumlah pengguna	Waktu Sinkron		Rerata
	Uji Coba	Restful API	Restful API
10	I	2.1	2.3
	II	2.5	
	III	2.3	
	IV	2.6	
	V	2.2	
20	I	3.9	3.9
	II	3.8	
	III	3.9	
	IV	4	
	V	3.9	
30	I	4.9	4.6
	II	4.5	
	III	4.8	
	IV	4.5	
	V	4.5	
Rerata			10.8

KESIMPULAN

Berdasarkan hasil analisa dan uji aplikasi yang dilakukan dapat disimpulkan bahwa, Aplikasi membantu proses tes sertifikasi Pada perusahaan, Aplikasi sudah memanfaatkan RestFul API sebagai sarana penghubung antara aplikasi dengan database, Pada pengujian fungsionalitas didapatkan hasil 100% fungsi pada aplikasi berjalan dengan baik, Pada pengujian Whitebox pada aplikasi didapatkan hasil persentase keberhasilan sebesar 100%. Waktu sinkronisasi data dengan 10 sampai 30 pengguna untuk 5 kali pengujian diperoleh waktu 10,8 detik.

REFERENSI

[1] D. Wijnarko and W. K. Dewanto, "PENGUNAAN MOODLEBILE PADA M-LEARNING."

[2] C. Agustina and . S., "Rancang Bangun Aplikasi Elektronik Transportasi Bandara Yogyakarta International Airport Berbasis Mobile Menggunakan Android Studio," *J. Teknol. Inf. dan Terap.*, vol. 7, no. 2, pp. 113–117, 2020,

doi: 10.25047/jtit.v7i2.136.

[3] M. Tarigan, "Pengembangan Sistem Stock Opname Berbasis Mobile Application Using SDLC Methode," vol. 8, no. 1, pp. 52–59, 2021.

[4] P. S. Pendingin, "ALGORITMA RANDOM PADA COMPUTER BASED TEST," pp. 157–163, 2010.

[5] U. S. Masuk and P. Tinggi, "PERANCANGAN APLIKASI UJIAN SARINGAN MASUK PERGURUAN TINGGI SECARA ONLINE BERBASIS ANDROID," vol. 15, no. 3, pp. 217–223, 2016.

[6] J. Pengabdian and P. Masyarakat, "IMPLEMENTASI COMPUTER BASED TEST (CBT) DI SEKOLAH MENENGAH," vol. 5458.

[7] F. Huzaimah and D. Irfan, "RANCANG BANGUN APLIKASI UJIAN ONLINE PRA KOMPRES BERBASIS ANDROID Fazriani Huzaimah 1) , Dedy Irfan 2) 1," vol. 6, no. 2, 2018.

[8] A. Kusumaningrum, H. Sajati, and D. Anarianto, "Rest and Soap Comparison on Web Service Technology for Android Based Data Services," *Conf. Senat. STT Adisutjipto Yogyakarta*, vol. 5, pp. 335–344, 2019, doi: 10.28989/senatik.v5i0.355.

[9] R. Choirudin and A. Adil, "Implementasi Rest Api Web Service dalam Membangun Aplikasi Multiplatform untuk Usaha Jasa," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 18, no. 2, pp. 284–293, 2019, doi: 10.30812/matrik.v18i2.407.

[10] Hasanuddin, H. Asgar, and B. Hartono, "Rancang Bangun Rest Api Aplikasi Weshare Sebagai Upaya Mempermudah Pelayanan Donasi Kemanusiaan," *J. Inform. Teknol. dan Sains*, vol. 4, no. 1, pp. 8–14, 2022, doi: 10.51401/jinteks.v4i1.1474.

[11] K. P. T. Xyz, "RESTFul Web b Service Untuk Sistem m Pencatatan Transaksi St Studi," vol. 2, no. April, 2016.

[12] T. W. A. Isnaini Rodiyah, Totok Totok, "sertifikasi.pdf," *Cakrawala*, vol. 6, pp. 157–169, 2012, doi: https://doi.org/10.32781/cakrawala.v6i2.152.

[13] I. Sommerville, *Software Engineering (9th ed.; Boston, Ed.). Massachusetts: Pearson Education*. 2011.