**PENGUJIAN PERANGKAT LUNAK**

**SISTEM INFORMASI ASSET KANTOR**

**WHITE BOX TESTING**



Oleh :

**Fardan Syahrul Hikam**

**151811513043**

**PROGRAM STUDI D3 SISTEM INFORMASI**

**DEPARTEMEN TEKNIK**

**FAKULTAS VOKASI**

**UNIVERSITAS AIRLANGGA**

**2020**

# Pengantar White Box Testing

White Box Testing adalah salah satu cara untuk menguji suatu aplikasi atau software dengan melihat modul untuk memeriksa dan menganalisis kode program ada yang salah atau tidak. Jika modul ini dan telah diproduksi dalam output yang tidak memenuhi persyaratan, kode akan dikompilasi ulang dan diperiksa lagi sampai mencapai apa yang diharapkan singkatnya White Box Testing ini menguji dengan cara melihat Pure Code dari suatu aplikai/software yang diuji tanpa memperdulikan Tampilan atau UI dari aplikasi tersebut..

Teknik White-box Testing

* Basis Path Testing

Basis path testing merupakan metode yang memungkinkan perancang testcase untuk membuat pengukuran kompleksitas logikal dari rancangan prosedural dan menggunakan pengukuran ini sebagai panduan untuk mendefinisikan himpunan basis dari jalur eksekusi. Test case yang dibuat untuk menguji himpunan basis dijamin akan mengeksekusi setiap statement di dalam program sekurangnya sekali pada saat pengujian

* Flow Graph

Flow graph merupakan notasi sederhana untuk merepresentasi control flow.

* Cyclomatic Complexity

Cyclomatic complexity digunakan untuk mengetahui jumlah jalur yang perlu dicari. Cyclomatic complexity adalah metric software yang menyediakan ukuran kuantitatif dari kompleksitas logikal program. Nilai yang dihitung bagi cyclomatic complexity menentukan  jumlah jalur-jalur yang independen dalam kumpulan basis suatu program dan memberikan jumlah tes minimal yang harus dilakukan untuk memastikan bahwa semua pernyataan telah dieksekusi sekurangnya satu kali.

Cyclomatic complexity mempunyai fondasi dalam teori graph dan dapat dihitung dengan satu dari tiga cara :

1. Jumlah region sama dengan cyclomatic complexity.
2. Cyclomatic complexity, V(G), untuk sebuah flow graph, G, didefnisikan sebagai: V(G) = E – N + 2 E adalah jumlah edge pada flow graph, dan N adalah jumlah node pada flow graph.
3. Cyclomatic complexity, V(G), untuk flow graph, G, juga didefinisikan sebagai: V(G) = P + 1 P adalah jumlah predicate nodes yang terdapat pada flow graph G.

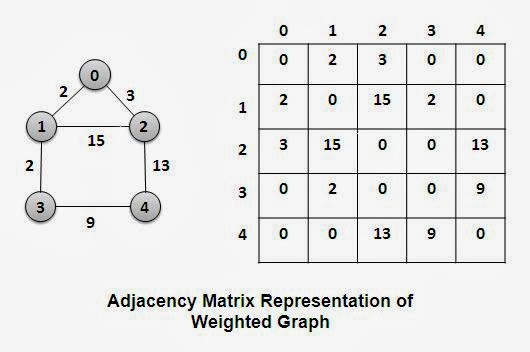
* Graph Matrix

Graph Matrix adalah matrix segi empat dengan jumlah baris dan kolom sama dengan jumlah node, dan identifikasi baris dan kolom sama dengan identifikasi node, serta isi data adalah keberadaan penghubung antar node (edges).

Beberapa properti yang dapat ditambahkan sebagai pembobotan pada koneksi antar node di dalam graph matrix, sebagai berikut:

* 1. Kemungkinan jalur (edge) yang akan dilalui / dieksekusi.
  2. Waktu proses yang diharapkan pada jalur selama proses transfer dilakukan.
  3. Memori yang dibutuhkan selama proses transfer dilakukan pada jalur.
  4. Sumberdaya (resources) yang dibutuhkan selama proses transfer dilakukan pada jalur.

Dalam pembuatan graph matrix dapat dipermudah dengan melihat flow graph. Contoh sederhana pemakaian graph matrix sebagai berikut :



Gambar 1 Flow Graph     Gambar 2 Graph Matrix

Kelemahan White Box-Testing:

1. Sangat mahal untuk dilakukan karena membutuhkan tester yang terampil untuk melakukan pengujian.
2. Pada perangkat lunak yang jenisnya besar, metode white box testing ini dianggap boros karena melibatkan banyak sumberdaya untuk melakukannya.
3. Tidak mempedulikan Tampilan UI aplikasinya.

# Gambaran Aplikasi Aset Kantor

Gambaran Aplikasi Aset Kantor Sistem informasi asset kantor adalah aplikasi inventaris yang digunakan untuk mencatat barang asset kantor dan sejarah perawatannya.

Aplikasi ini memiliki 2 peran yaitu :

* Admin

Bertugas membuat, menghapus, dan mengubah data pengguna

* Operator

Bertugas mengelola data barang, perawatan, dan mengganti sandi milik sendiri

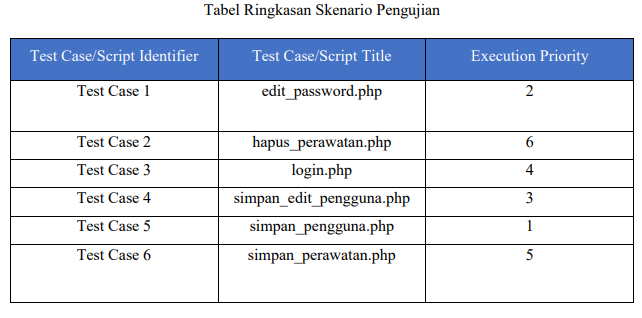
# Ringkisan Kasus Uji

Dalam pengujian Sistem informasi Aset Kantor ini saya melakukan pengujian dengan menggunkan jenis whitebox :

1. Basis Path

2. Complexity Cyclometic

3. Graph Matrix



# Rincian Kasus Uji

Berikut adalah speseifikasi setiap tes skenario dalam pengujian perangkat lunak sistem informasi Aset Kantor.

## Edit\_password.php

|  |  |
| --- | --- |
| A0 | <?php |
| A1 | session\_start(); |
| A2 | $id\_pengguna = $\_SESSION['id\_pengguna']; |
| A3 | $sandi = $\_POST['sandi']; |
| A4 | $conn = new mysqli('localhost', 'root', '', 'aset\_kantor'); |
| A5 | if ($conn->connect\_error) { |
| A6 | die("Connection failed: " . $conn->connect\_error); |
| A7 | } |
| A8 | $sql = " |
| A9 | UPDATE `pengguna` |
| A10 | SET `sandi`='$sandi' |
| A11 | WHERE `id\_pengguna`='$id\_pengguna' |
| A12 | "; |
| A13 | $result = $conn->query($sql); |
| A14 | if($result){ |
| A15 | echo "Password Berhasil Diupdate"; |
| A16 | } |
| A17 | ?> |

* + 1. Basis Path



* + 1. Graph Matrix
       1. Tabel Graph Matrix

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A4** | **A5** | **A6** | **A7- A13** | **A14** | **A15** | **A16** | **A17** |
| **A0- A4** |  | a |  |  |  |  |  |  |
| **A5** |  |  | i | b |  |  |  |  |
| **A6** |  |  |  |  |  |  |  | j |
| **A7- A13** |  |  |  |  | c |  |  |  |
| **A14** |  |  |  |  |  | g | d |  |
| **A15** |  |  |  |  |  |  |  | h |
| **A16** |  |  |  |  |  |  |  | f |
| **A17** |  |  |  |  |  |  |  |  |

* + - 1. Tabel Conection Matrix

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A4** | **A5** | **A6** | **A7-A13** | **A14** | **A15** | **A16** | **A17** | **connection** |
| **A0- A4** |  | 1 |  |  |  |  |  |  | 1-1 = 0 |
| **A5** |  |  | 1 | 1 |  |  |  |  | 2-1 = 1 |
| **A6** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A7- A13** |  |  |  |  | 1 |  |  |  | 1-1 = 0 |
| **A14** |  |  |  |  |  | 1 | 1 |  | 2-1 = 1 |
| **A15** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A16** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A17** |  |  |  |  |  |  |  |  | 0 |
| **TOTAL CONNECTION** | | | | | | | | | 2 |

* + 1. Complexity Cyclometic

* + 1. Independent Path

1. A0-A4 – A5 – A7 – A8– A9 – A10 – A11 – A12 – A13 – A14 – A16 – A17
2. A0-A4 – A5 – A6 – A7 – A8 – A9 – A10 – A11 – A12 – A13 – A14 – A16 – A17
3. A0-A4 – A5 – A6 – A7 – A8 – A9 – A10 – A11 – A12 – A13 – A14 – A15 – A16 – A17
   * 1. Test Case
4. 1. A(5) = False
   2. A(14) = False
5. 1. A(5) = True
   2. A(14) = False
6. 1. A(5) = False
   2. A(14) = True

1. 1. A(5) = True
   2. A(14) = True

## Hapus\_perawatan.php

|  |  |
| --- | --- |
| A0 | <?php |
| A1 | $id\_perawatan = $\_GET['id\_perawatan']; |
| A2 | $conn = new mysqli('localhost', 'root', '', 'aset\_kantor'); |
| A3 | if ($conn->connect\_error) { |
| A4 | die("Connection failed: " . $conn->connect |
| A5 | } |
| A6 | $sql = "DELETE FROM `perawatan` WHERE `id\_perawatan`='$id\_perawatan' "; |
| A7 | $result = $conn->query($sql); |
| A8 | if($result){ |
| A9 | echo "Hapus data Berhasil"; |
| A10 | } |
| A11 | else { |
| A12 | echo "Hapus Data Gagal"; |
| A13 | } |
| A14 | ?> |

* + 1. Basis Path



* + 1. Graph Matrix
       1. Tabel Graph Matrix

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A2** | **A3** | **A4** | **A5- A7** | **A8** | **A9** | **A10-A12** | **A13** | **A14** |
| **A0- A2** |  | a |  |  |  |  |  |  |  |
| **A3** |  |  | i | b |  |  |  |  |  |
| **A4** |  |  |  |  |  |  |  | j |  |
| **A5- A7** |  |  |  |  | c |  |  |  |  |
| **A8** |  |  |  |  |  | g | d |  |  |
| **A9** |  |  |  |  |  |  |  | h |  |
| **A10-A12** |  |  |  |  |  |  |  | f |  |
| **A13** |  |  |  |  |  |  |  |  | k |
| **A14** |  |  |  |  |  |  |  |  |  |

* + - 1. Tabel Connection Matrix

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A2** | **A3** | **A4** | **A5- A7** | **A8** | **A9** | **A10-A12** | **A13** | **A14** | **Connection** |
| **A0- A2** |  | 1 |  |  |  |  |  |  |  | 1-1 = 0 |
| **A3** |  |  | 1 | 1 |  |  |  |  |  | 2-1 = 1 |
| **A4** |  |  |  |  |  |  |  | 1 |  | 1-1 = 0 |
| **A5- A7** |  |  |  |  | 1 |  |  |  |  | 1-1 = 0 |
| **A8** |  |  |  |  |  | 1 | 1 |  |  | 2-1 = 1 |
| **A9** |  |  |  |  |  |  |  | 1 |  | 1-1 = 0 |
| **A10-A12** |  |  |  |  |  |  |  | 1 |  | 1-1 = 0 |
| **A13** |  |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A14** |  |  |  |  |  |  |  |  |  | 0 |
| **Total Connection** | | | | | | | | | | 2 |

* + 1. Complexity Cyclometic
    2. Independent Path

1. A0-A2 – A3 – A6 – A7– A8 – A13 – A14
2. A0-A2 – A3 – A4 – A5 – A6 – A7– A8 – A13 – A14
3. A0-A2 – A3 – A4 – A5 – A6 – A7– A8 – A9 – A10 – A11 – A12 – A13 – A14
   * 1. Test Case

I.

1. A(3) = False

2. A(8) = False

II.

1. A(3) = False

2. A(8) = True

III.

1. A(3) = True

2. A(8) = False

IV.

1. A(3) = True

2. A(8) = True

## Login.php

|  |  |
| --- | --- |
| A0 | <?php |
| A1 | $username = $\_POST['username']; |
| A2 | $password = $\_POST['password']; |
| A3 | $conn = new mysqli('localhost', 'root', '', 'aset\_kantor'); |
| A4 | if ($conn->connect\_error) { |
| A5 | die("Connection failed: " . $conn->connect\_error); |
| A6 | } |
| A7 | $sql = "SELECT `id\_pengguna`, `login`, `sandi`, `peran` FROM `pengguna` WHERE `login`='$username' and `sandi`='$password' "; |
| A8 | $result = $conn->query($sql); |
| A9 | $akses =null; |
| A10 | if ($result->num\_rows > 0) { |
| A11 | session\_start(); |
| A12 | while($row = $result->fetch\_assoc()) { |
| A13 | $\_SESSION['id\_pengguna'] = $row['id\_pengguna']; |
| A14 | $\_SESSION['peran'] = $row["peran"]; |
| A15 | $\_SESSION['login'] = $row["login"]; |
| A16 | if($row["peran"]=='0'){ |
| A17 | header('Location: v\_halaman\_admin.php'); |
| A18 | } |
| A19 | else if($row["peran"]=='1'){ |
| A20 | header('Location: v\_halaman\_operator.php'); |
| A21 | } |
| A22 | else { |
| A23 | echo "username salah"; |
| A24 | } |
| A25 | } |
| A26 | } else { |
| A27 | echo "0 results"; |
| A28 | } |
| A29 | ?> |

* + 1. Basis Path



* + 1. Graph Matrix
       1. Tabel Graph Matrix

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A3** | **A4** | **A5** | **A6- A9** | **A10** | **A11** | **A12** | **A13-A15** | **A16** | **A17** | **A18** | **A19** | **A20** | **A21-A23** | **A24** | **A25** | **A26-A28** | **A29** |
| **A0- A3** |  | a |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **A4** |  |  | u | b |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **A5** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | v |
| **A6- A9** |  |  |  |  | c |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **A10** |  |  |  |  |  | d |  |  |  |  |  |  |  |  |  |  | g |  |
| **A11** |  |  |  |  |  |  | e |  |  |  |  |  |  |  |  |  |  |  |
| **A12** |  |  |  |  |  |  |  | f |  |  |  |  |  |  |  |  |  |  |
| **A13-A15** |  |  |  |  |  |  |  |  | g |  |  |  |  |  |  |  |  |  |
| **A16** |  |  |  |  |  |  |  |  |  | q | h |  |  |  |  |  |  |  |
| **A17** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | r |  |  |  |
| **A18** |  |  |  |  |  |  |  |  |  |  |  | i |  |  |  |  |  |  |
| **A19** |  |  |  |  |  |  |  |  |  |  |  |  | j | l |  |  |  |  |
| **A20** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | k |  |  |  |
| **A21-A23** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | m |  |  |  |
| **A24** |  |  |  |  |  |  | p |  |  |  |  |  |  |  |  | n |  |  |
| **A25** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | o |
| **A26-A28** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | t |  |
| **A29** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

* + - 1. Tabel Connection Matrix

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A3** | **A4** | **A5** | **A6- A9** | **A10** | **A11** | **A12** | **A13-A15** | **A16** | **A17** | **A18** | **A19** | **A20** | **A21-A23** | **A24** | **A25** | **A26-A28** | **A29** | **Connection** |
| **A0- A3** |  | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1-1 = 0 |
| **A4** |  |  | 1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 2-1 = 1 |
| **A5** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A6- A9** |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  | 1-1 = 0 |
| **A10** |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  | 1 |  | 2-1 = 1 |
| **A11** |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  | 1-1 = 0 |
| **A12** |  |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  | 1-1 = 0 |
| **A13-A15** |  |  |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  | 1-1 = 0 |
| **A16** |  |  |  |  |  |  |  |  |  | 1 | 1 |  |  |  |  |  |  |  | 2-1 = 1 |
| **A17** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  | 1-1 = 0 |
| **A18** |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  |  |  | 1-1 = 0 |
| **A19** |  |  |  |  |  |  |  |  |  |  |  |  | 1 | 1 |  |  |  |  | 2-1 = 1 |
| **A20** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  | 1-1 = 0 |
| **A21-A23** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  | 1-1 = 0 |
| **A24** |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  | 1 |  |  | 2-1 = 1 |
| **A25** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A26-A28** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  | 1-1 = 0 |
| **A29** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 0 |
| **Total Connection** | | | | | | | | | | | | | | | | | | | 5 |

* + 1. Complexity Cyclometic

6

* + 1. Independent Path

1. A0-A3 – A4 – A7 – A8 – A9 – A29
2. A0-A3 – A4 – A5 – A6 – A7 – A8 – A9 – A28 – A29
3. A0-A3 – A4 – A5 – A6 – A7 – A8 – A9 – A10 – A11 – A12 – A26 – A27 – A28 – A29
4. A0-A3 – A4 – A5 – A6 – A7 – A8 – A9 – A10 – A11 – A12 – A13 – A14 – A15 – A16 – A26 – A27 – A28 – A29
5. A0-A3 – A4 – A5 – A6 – A7 – A8 – A9 – A10 – A11 – A12 – A13 – A14 – A15 – A16 – A17 – A18 – A19 – A26 – A27 – A28 – A29
6. A0-A3 – A4 – A5 – A6 – A7 – A8 – A9 – A10 – A11 – A12 – A13 – A14 – A15 – A16 – A17 – A18 – A19 – A20 – A21 – A22 – A23 – A24 – A25 – A26 – A27 – A28 – A29
   * 1. Test Case
8. A(4) = False
9. A(9) = False
10. A(12) = False
11. A(16) = False
12. A(19) = False
14. A(4) = True
15. A(9) = False
16. A(12) = False
17. A(16) = False
18. A(19) = False
20. A(4) = True
21. A(9) = True
22. A(12) = False
23. A(16) = False
24. A(19) = False
26. A(4) = True
27. A(9) = True
28. A(12) = True
29. A(16) = False
30. A(19) = False
32. A(4) = True
33. A(9) = True
34. A(12) = True
35. A(16) = True
36. A(19) = False
38. A(4) = True
39. A(9) = True
40. A(12) = True
41. A(16) = True
42. A(19) = True
44. A(4) = True
45. A(9) = True
46. A(12) = True
47. A(16) = True
48. A(19) = False
50. A(4) = True
51. A(9) = True
52. A(12) = True
53. A(16) = False
54. A(19) = False
56. A(4) = True
57. A(9) = True
58. A(12) = False
59. A(16) = False
60. A(19) = False
62. A(4) = True
63. A(9) = False
64. A(12) = False
65. A(16) = False
66. A(19) = False

## Simpan\_edit\_pengguna.php

|  |  |
| --- | --- |
| A0 | <?php |
| A1 | $id\_pengguna = $\_POST['id\_pengguna']; |
| A2 | $login = $\_POST['login']; |
| A3 | $sandi = $\_POST['sandi']; |
| A4 | $optradio = $\_POST['optradio']; |
| A5 | $conn = new mysqli('localhost', 'root', '', 'aset\_kantor'); |
| A6 | if ($conn->connect\_error) { |
| A7 | die("Connection failed: " . $conn->connect\_error); |
| A8 | } |
| A9 | $sql = " |
| A10 | UPDATE `pengguna` |
| A11 | SET `login`='$login', |
| A12 | `sandi`='$sandi', |
| A13 | `peran`='$optradio' |
| A14 | WHERE `id\_pengguna`='$id\_pengguna' |
| A15 | "; |
| A16 | $result = $conn->query($sql); |
| A17 | if($result){ |
| A18 | echo "Data Berhasil Diubah"; |
| A19 | } |
| A20 | ?> |

* + 1. Basis Path



* + 1. Graph Matrix
       1. Tabel Graph Matrix

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A5** | **A6** | **A7** | **A8- A16** | **A17** | **A18** | **A19** | **A20** |
| **A0- A5** |  | a |  |  |  |  |  |  |
| **A6** |  |  | i | b |  |  |  |  |
| **A7** |  |  |  |  |  |  |  | j |
| **A8- A16** |  |  |  |  | c |  |  |  |
| **A17** |  |  |  |  |  | g | d |  |
| **A18** |  |  |  |  |  |  |  | h |
| **A19** |  |  |  |  |  |  |  | f |
| **A20** |  |  |  |  |  |  |  |  |

* + - 1. Tabel Connection Matrix

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A5** | **A6** | **A7** | **A8- A16** | **A17** | **A18** | **A19** | **A20** | **A0-A5** |
| **A0- A5** |  | 1 |  |  |  |  |  |  | 1-1 = 0 |
| **A6** |  |  | 1 | 1 |  |  |  |  | 2-1 = 1 |
| **A7** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A8- A16** |  |  |  |  | 1 |  |  |  | 1-1 = 0 |
| **A17** |  |  |  |  |  | 1 | 1 |  | 2-1 = 1 |
| **A18** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A19** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A20** |  |  |  |  |  |  |  |  | 0 |
| **Total Connection** | | | | | | | | | **2** |

* + 1. Complexity Cyclometic
    2. Independent Path

1. A0-A5 – A6 – A9-A16 – A17 – A19 – A20
2. A0-A5 – A6 – A7 – A8 – A9-A16 – A17 – A19 – A20
3. A0-A5 – A6 – A7 – A8 – A9-A16 – A17 – A18 – A19 – A20
   * 1. Test Case

1. A(6) = False

2. A(17) = False

1. 1. A(6) = False
   2. A(17) = True
2. 1. A(6) = True
   2. A(17) = False
3. 1. A(6) = True
   2. A(17) = True

## Simpan\_pengguna.php

|  |  |
| --- | --- |
| A0 | <?php |
| A1 | $id\_pengguna = $\_POST['id\_pengguna']; |
| A2 | $login = $\_POST['login']; |
| A3 | $sandi = $\_POST['sandi']; |
| A4 | $optradio = $\_POST['optradio']; |
| A5 | $conn = new mysqli('localhost', 'root', '', 'aset\_kantor'); |
| A6 | if ($conn->connect\_error) { |
| A7 | die("Connection failed: " . $conn->connect\_error); |
| A8 | } |
| A9 | $sql = "INSERT INTO `pengguna`(`id\_pengguna`, `login`, `sandi`, `peran`) |
| A10 | VALUES ('$id\_pengguna','$login','$sandi','$optradio') "; |
| A11 | $result = $conn->query($sql); |
| A12 | if($result){ |
| A13 | echo "Data Berhasil Ditambah"; |
| A14 | } |
| A15 | ?> |

* + 1. Basis Path



* + 1. Graph Matrix
       1. Tabel Graph Matrix

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A5** | **A6** | **A7** | **A8- A11** | **A12** | **A13** | **A14** | **A15** |
| **A0- A5** |  | a |  |  |  |  |  |  |
| **A6** |  |  | i | b |  |  |  |  |
| **A7** |  |  |  |  |  |  |  | j |
| **A8- A11** |  |  |  |  | c |  |  |  |
| **A12** |  |  |  |  |  | g | d |  |
| **A13** |  |  |  |  |  |  |  | h |
| **A14** |  |  |  |  |  |  |  | f |
| **A15** |  |  |  |  |  |  |  |  |

* + - 1. Tabel Connection Matrix

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A5** | **A6** | **A7** | **A8- A11** | **A12** | **A13** | **A14** | **A15** | **Connection** |
| **A0- A5** |  | 1 |  |  |  |  |  |  | 1-1 = 0 |
| **A6** |  |  | 1 | 1 |  |  |  |  | 2-1 = 1 |
| **A7** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A8- A11** |  |  |  |  | 1 |  |  |  | 1-1 = 0 |
| **A12** |  |  |  |  |  | 1 | 1 |  | 2-1 = 1 |
| **A13** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A14** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A15** |  |  |  |  |  |  |  |  |  |
| **Total Connection** | | | | | | | | | **2** |

* + 1. Complexity Cyclometic

* + 1. Independent Path

1. A0-A5 – A6 – A9-A11 – A12 – A15
2. A0-A5 – A6 – A7 – A8 – A9-A11– A12 – A15
3. A0-A5 – A6 – A7 – A8 – A9-A11– A12 – A13 – A14 – A15
   * 1. Test Case
4. 1. A(6) = False
   2. A(12) = False
6. A(6) = False
7. A(12) = True
8. 1. A(6) = True
   2. A(12) = False
9. 1. A(6) = True
   2. A(12) = True

## Simpan\_perawatan.php

|  |  |
| --- | --- |
| A0 | <?php |
| A1 | $id\_perawatan = $\_POST[‘id\_perawatan’]; |
| A2 | $barang = $\_POST[‘barang’]; |
| A3 | $tgl\_perawatan = $\_POST[‘tgl\_perawatan’]; |
| A4 | $keterangan = $\_POST[‘keterangan’]; |
| A5 | $biaya\_perawatan = $\_POST[‘biaya\_perawatan’]; |
| A6 | $opstatus = $\_POST[‘opstatus’]; |
| A7 | $conn = new mysqli(‘localhost’, ‘root’, ‘’, ‘aset\_kantor’); |
| A8 | if ($conn->connect\_error) { |
| A9 | die(“Connection failed: “ . $conn->connect\_error); |
| A10 | } |
| A11 | session\_start(); |
| A12 | $pengguna = $\_SESSION[‘id\_pengguna’]; |
| A13 | $sql = “INSERT INTO `perawatan` |
| A14 | (`id\_perawatan`, `id\_barang`, `tgl\_perawatan`, `keterangan`, `biaya\_perawatan`, `status`, `id\_pengguna`) |
| A15 | VALUES |
| A16 | (‘$id\_perawatan’,’$barang’,’$tgl\_perawatan’,’$keterangan’,’$biaya\_perawatan’,’$opstatus’,’$pengguna’) “; |
| A17 | $result = $conn->query($sql); |
| A18 | if($result){ |
| A19 | echo “Data Berhasil di Simpan”; |
| A20 | } |
| A21 | >? |

* + 1. Basis Path



* + 1. Graph Matrix
       1. Tabel Graph Matrix

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A7** | **A8** | **A9** | **A10- A17** | **A18** | **A19** | **A20** | **A21** |
| **A0- A7** |  | a |  |  |  |  |  |  |
| **A8** |  |  | i | b |  |  |  |  |
| **A9** |  |  |  |  |  |  |  | j |
| **A10- A17** |  |  |  |  | c |  |  |  |
| **A18** |  |  |  |  |  | g | d |  |
| **A19** |  |  |  |  |  |  |  | h |
| **A20** |  |  |  |  |  |  |  | f |
| **A21** |  |  |  |  |  |  |  |  |

* + - 1. Tabel Connection Matrix

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **To From** | **A0-A5** | **A6** | **A7** | **A8- A16** | **A17** | **A18** | **A19** | **A20** | **A0-A5** |
| **A0- A5** |  | 1 |  |  |  |  |  |  | 1-1 = 0 |
| **A6** |  |  | 1 | 1 |  |  |  |  | 2-1 = 1 |
| **A7** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A8- A16** |  |  |  |  | 1 |  |  |  | 1-1 = 0 |
| **A17** |  |  |  |  |  | 1 | 1 |  | 2-1 = 1 |
| **A18** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A19** |  |  |  |  |  |  |  | 1 | 1-1 = 0 |
| **A20** |  |  |  |  |  |  |  |  |  |
| **Total Connection** | | | | | | | | | **2** |

* + 1. Complexity Cyclometic
    2. Independent Path

1. A0-A7 – A8 – A11-A17 – A18 – A21
2. A0-A7 – A8 – A9 – A10 – A11-A17 – A18 – A21
3. A0-A7 – A8 – A9 – A10 – A11-A17 – A18 – A19 – A20 – A21
   * 1. Test Case

I.

1. A(6) = False

2. A(12) = False

II.

1. A(6) = False

2. A(12) = True

III.

1. A(6) = True

2. A(12) = False

IV.

1. A(6) = True

2. A(12) = True