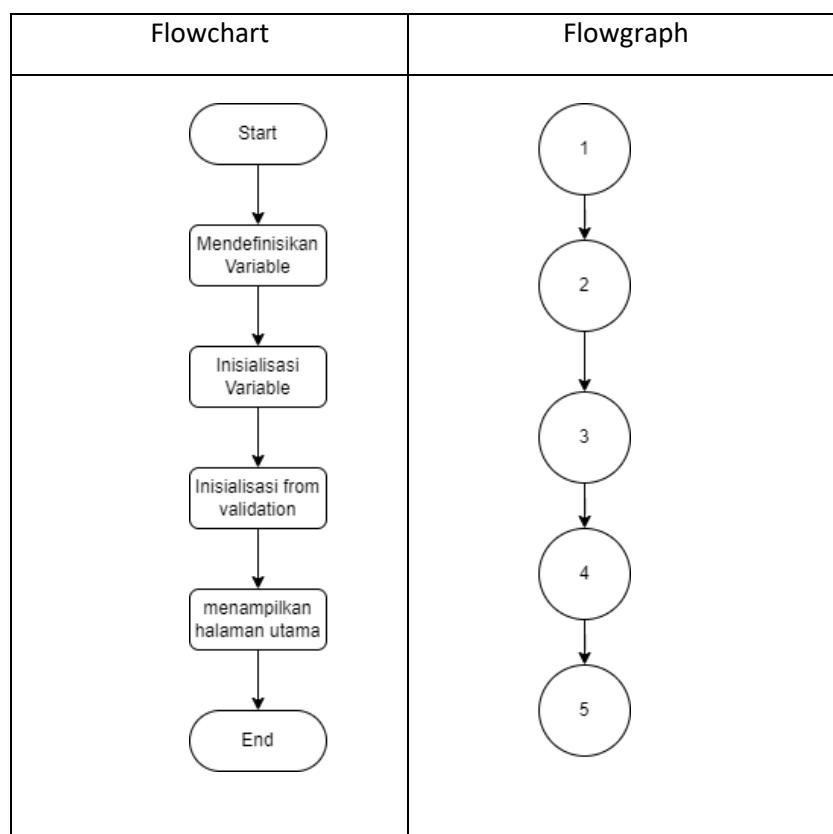


## WHITEBOX TESTING

### A. Pengujian White Box Pada Login

```
public function __construct()
{
    parent::__construct();
    $this->load->helper(['url', 'form', 'sia', 'tgl_indo']);
    $this->load->library(['session', 'form_validation']);
    $this->load->model('User_model', 'user', true);
}
```

1. Flowchart ini dibuat agar dapat mengonversikannya ke dalam bentuk flowgraph. Secara umum Flowchart Form Login yang telah ditampilkan.



2. Cyclomatic Complexity

Untuk menghitung menggunakan formula  $V(G) = E - N + 2$ . Berdasarkan penerapan flowgraph maka dapat diketahui nilainya sebagai berikut:

E: 4

N: 5

Masukan ke dalam formula sebagai berikut:

$$V(G) = E - N + 2$$

$$V(G) = 4 - 5 + 2$$

$$V(G) = 1$$

### 3. Region Testing

Jumlah Region = Jumlah kurva tertutup + 1(region terluar)

Jumlah Region = 0 + 1 = 1

### 4. Independen Path

Dari perhitungan cyclomatic didapatkan 1 jalur independent yaitu

Path 1 = 1-2-3-4-5

### 5. Matriks

	1	2	3	4	5
1		1			
2			1		
3				1	
4					1
5					

### 6. Predicate node

$V(G) = \text{Jumlah predicate node} + 1$

$V(G) = 1 + 1 = 2$

### 7. Test Case

NO	Kasus Uji	Hasil Yang DiHarapkan	Hasil	Keterangan

1.	1-2-3-4-5	Ketika pengguna mengisi form input user dan sandi benar maka akan meju ke halaman utama	Sistem menampilkan halaman utama setelah login berhasil	Alur terlewati
----	-----------	---	---	----------------

## B. Pengujian White Box Dashboard dan Data Akun

Fungsi Index

- Procedure sort

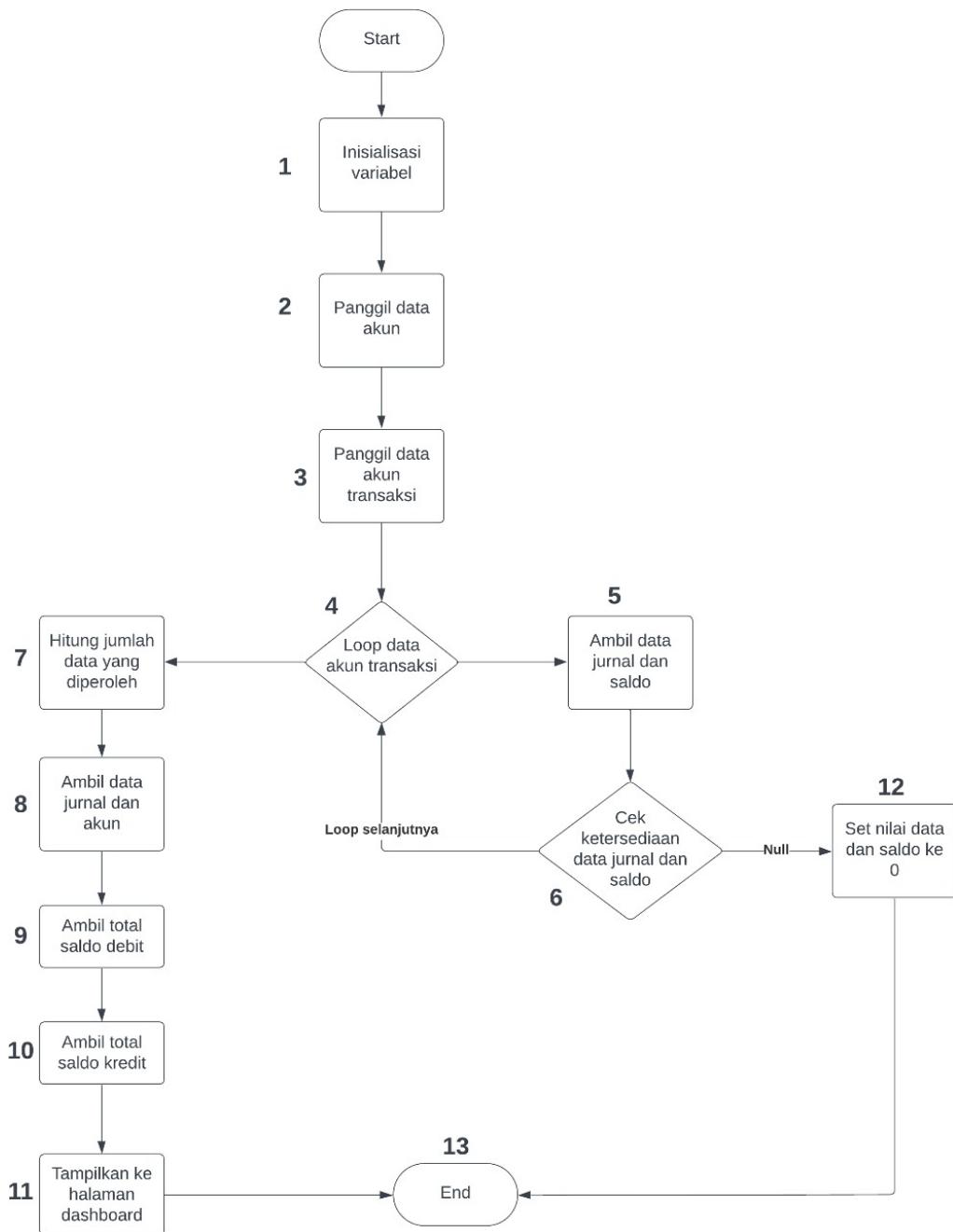
```
public function index(){
    $titleTag = 'Dashboard';
    $content = 'user/dashboard';
    $dataAkun = $this->akun->getAkun();
    $dataAkunTransaksi = $this->jurnal->getAkunInJurnal();

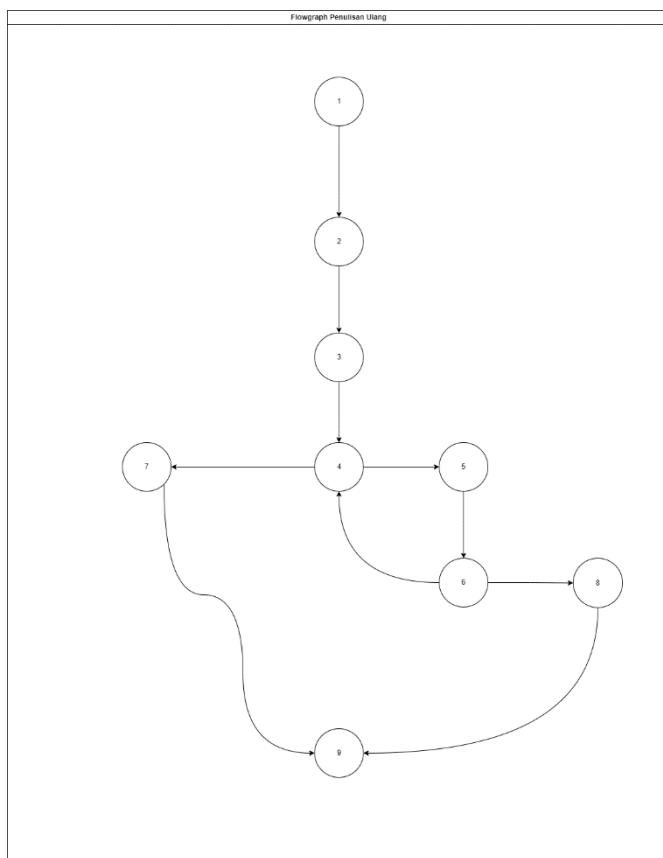
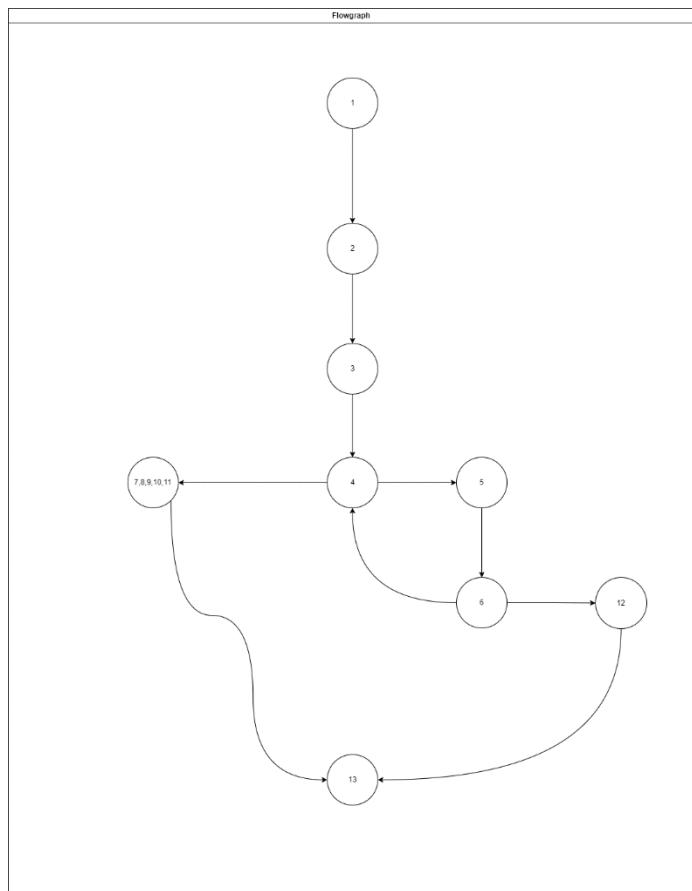
    foreach($dataAkunTransaksi as $row){
        $data[] = (array) $this->jurnal->getJurnalByNoReff($row->no_reff);
        $saldo[] = (array) $this->jurnal->getJurnalByNoReffSaldo($row-
>no_reff);
    }

    $jumlah = count($data);

    $journals = $this->jurnal->getJurnalJoinAkun();
    $totalDebit = $this->jurnal->getTotalSaldo('debit');
    $totalKredit = $this->jurnal->getTotalSaldo('kredit');
    $this->load-
>view('template',compact('content','dataAkun','titleTag','journals','totalDebit
','totalKredit','jumlah','data','saldo','dataAkunTransaksi'));
}
```

- Flowchart & Flowgraph





- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 10 - 9 + 2 = 3$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$\text{Jumlah region} = 2 + 1 = 3$$

- Independent Path

$$\text{Path 1} = 1-2-3-4-5-6-8-9$$

$$\text{Path 2} = 1-2-3-4-5-4-7-9$$

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5	6	7	8	9
1		1							
2			1						
3				1					
4					1		1		
5						1			
6				1				1	
7									1
8									1
9									

- Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 0 + 1 = 1$$

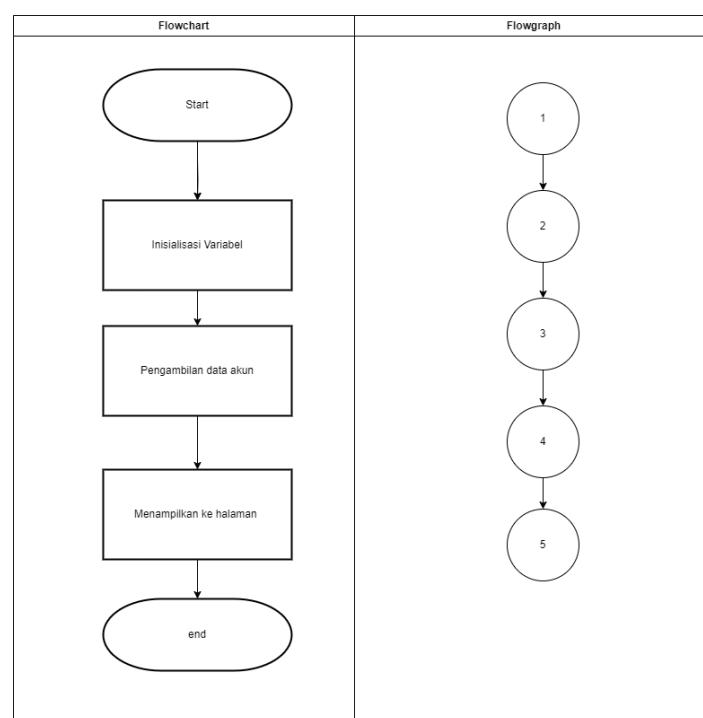
- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan

1	1-2-3- 4-5-6- 8-9	Pada menu halaman dashboard ketika data jurnal dan saldo tidak ada, maka set data saldo ke 0	Berhasil	Alur terlewati
2	1-2-3- 4-5-4- 7-9	Pada menu halaman dashboard ketika data jurnal dan saldo ada, maka data tersebut akan dihitung sesuai jumlah data yang diperoleh.	Berhasil	Alur terlewati

- Fungsi Data akun
  - Procedure sort

```
public function dataAkun(){
    $content = 'user/data_akun';
    $titleTag = 'Data Akun';
    $dataAkun = $this->akun->getAkun();
    $this->load->view('template',compact('content','dataAkun','titleTag'));
}
```



- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 4 - 5 + 2 = 1$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$\text{Jumlah region} = 0 + 1 = 1$$

- Independent Path

$$\text{Path 1} = 1-2-3-4-5$$

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5
1		1			
2			1		
3				1	
4					1
5					

- Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 0 + 1 = 1$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-3-4-5	Ketika pengguna klik menu Halaman data akun, maka sistem menampilkan data akun	Berhasil	Alur terlewati

## Fungsi Isnamakunthere

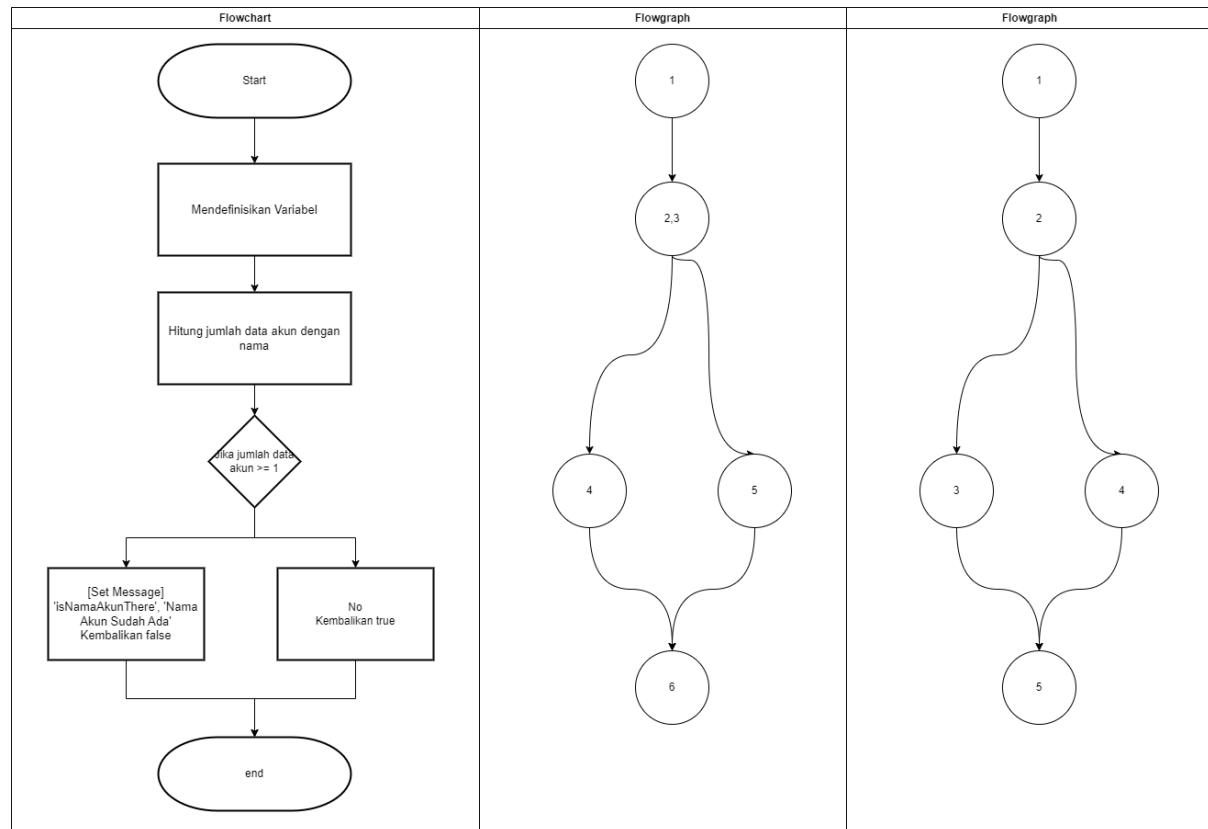
- Procedure sort

```
public function isNamaAkunThere($str){
```

```

$namaAkun = $this->akun->countAkunByNama($str);
if($namaAkun >= 1){
    $this->form_validation->set_message('isNamaAkunThere', 'Nama Akun Sudah Ada');
    return false;
}
return true;
}

```



- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 5 - 5 + 2 = 2$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

Jumlah region = 1 + 1 = 2

- Independent Path

Path 1 = 1-2-3-5

Path 2 = 1-2-4-5

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5
1		1			
2			1		
3				1	
4					1
5					

- Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

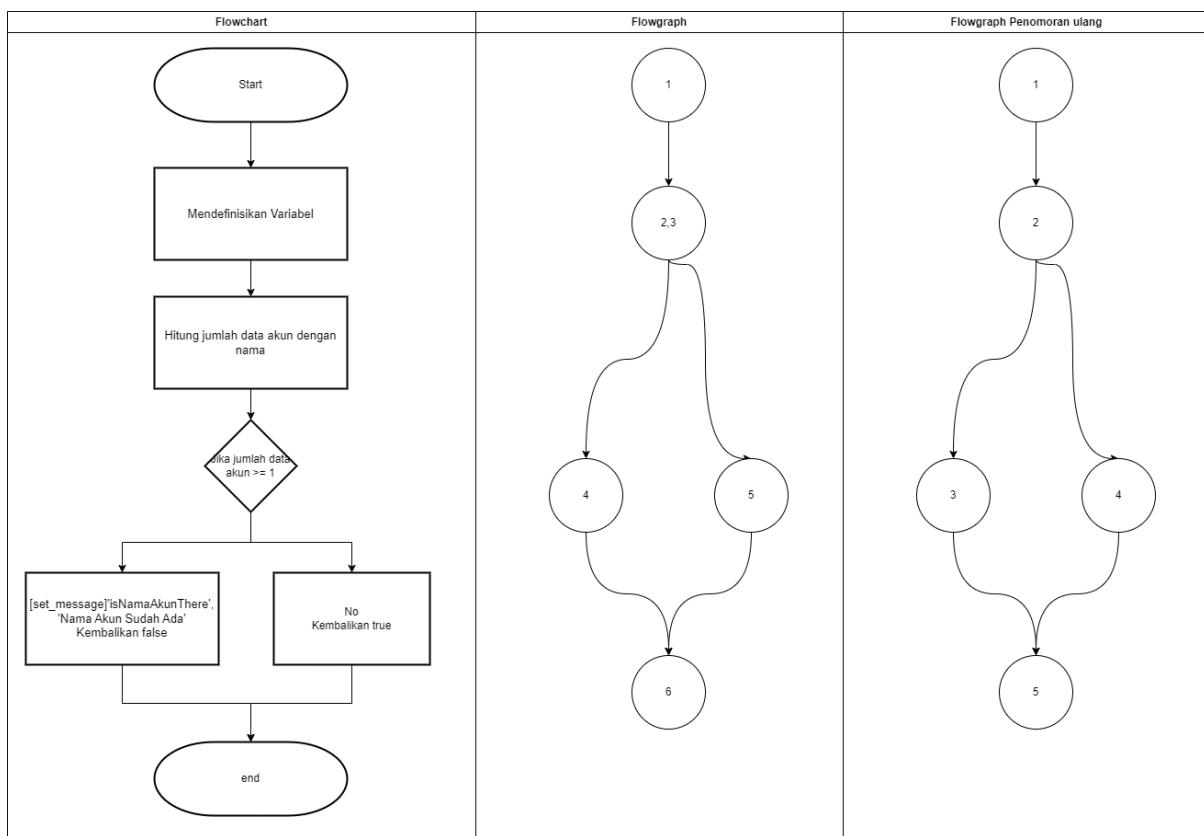
$$V(G) = 1 + 1 = 1$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-3-4-5	Ketika pemrosesan halaman data akun fungsi isNamaakunthere jika jumlah data akun lebih dari 1 akan menampilkan pesan akun sudah ada	Berhasil	Alur terlewati
2.	1-2-4-5	Ketika pemrosesan halaman data akun fungsi isNamaakunthere jika jumlah data akun tidak lebih dari 1 akan melanjutkan ke langkah selanjutnya	Berhasil	Alur terlewati

## Fungsi IsNoAkunThere

```
ublic function isNoAkunThere($str){
    $noAkun = $this->akun->countAkunByNoReff($str);
    if($noAkun >= 1){
        $this->form_validation->set_message('isNoAkunThere', 'No.Reff Sudah Ada');
        return false;
    }
    return true;
}
```



- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 5 - 5 + 2 = 2$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

Jumlah region = 1 + 1 = 2

- Independent Path

Path 1 = 1-2-3-5

Path 2 = 1-2-4-5

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5
1		1			
2			1		
3				1	
4					1
5					

- Predicate node

$V(G) = \text{Jumlah predicate node} + 1$

$$V(G) = 1 + 1 = 1$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-3-5	Ketika pemrosesan halaman data akun fungsi isNamaakunthere jika jumlah No. akun lebih dari 1 akan menampilkan pesan akun sudah ada	Berhasil	Alur terlewati
2.	1-2-4-5	Ketika pemrosesan halaman data akun fungsi isNamaakunthere jika jumlah No. akun tidak lebih dari 1 akan melanjutkan ke langkah selanjutnya	Berhasil	Alur terlewati

### Fungsi Create Akun

- Procedure Sort

```
public function createAkun(){
```

```

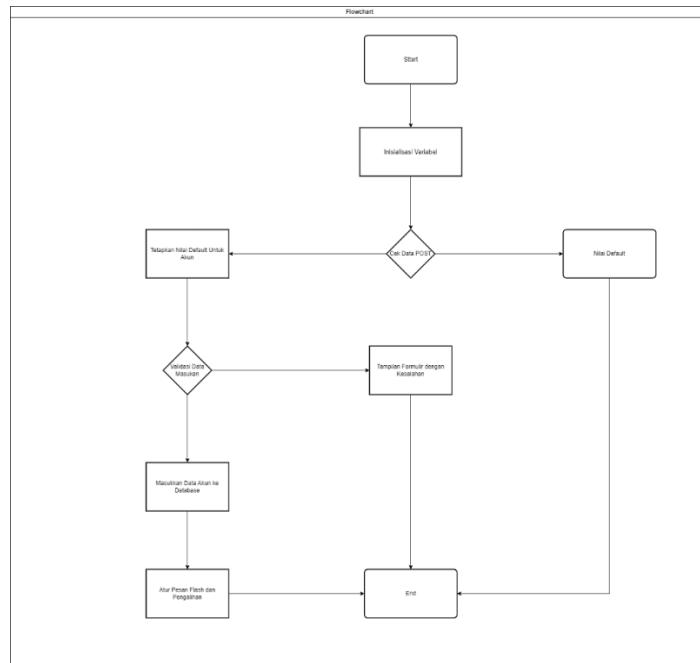
$title = 'Tambah';
$titleTag = 'Tambah Data Akun';
$action = 'data_akun/tambah';
$content = 'user/form_akun';

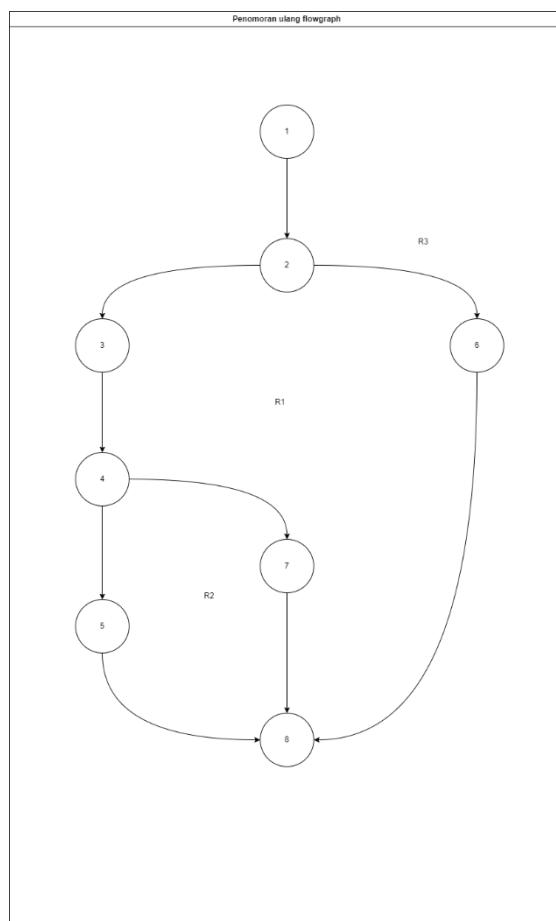
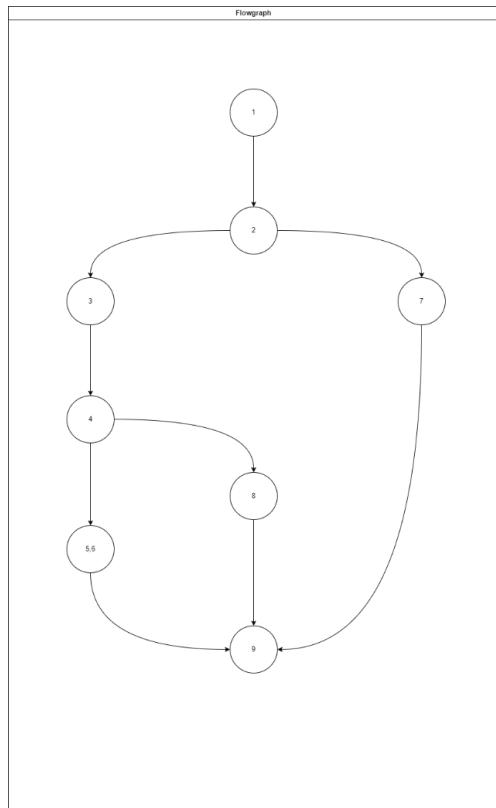
if(!$_POST){
    $data = (object) $this->akun->getDefaultValue();
} else{
    $data = (object) $this->input->post(null,true);
    $data->id_user = $this->session->userdata('id');
}

if(!$this->akun->validate()){
    $this->load->view('template',compact('content','title','action','data','titleTag'));
    return;
}

$this->akun->insertAkun($data);
$this->session->set_flashdata('berhasil','Data Akun Berhasil Di Tambahkan');
redirect('data_akun');
}

```





- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 9 - 8 + 2 = 3$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$2 + 1 = 3$$

- Independent path

$$\text{Path 1} = 1-2-7-9$$

$$\text{Path 2} = 1-2-3-4-8-9$$

$$\text{Path 3} = 1-2-3-4-5-6-9$$

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5	6	7	8
1		1						
2			1			1		
3				1				
4					1		1	
5								1
6								1
7								1
8								

- Predicate Node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 2 + 1 = 3$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan

1	1-2-7-9	Ketika pengguna tidak mengisi data pada form create akun, maka data jurnal akan diisi dengan data default	Kondisi if(!\$_POST) pada kode tidak dapat terjadi dalam praktik penggunaan normal karena UI tidak memungkinkan pengguna mengirimkan formulir tanpa mengisi data yang diperlukan. Oleh karena itu, kondisi ini tidak dapat diuji dalam kasus uji ini.	Alur tidak terlewati
2	1-2-3-4-8-9	Ketika pengguna mengisi data pada form create akun dengan data yang menyebabkan validasi gagal, maka sistem akan menampilkan pesan kesalahan	kondisi if(!\$this->akun->validate()) pada kode tidak dapat terjadi dalam praktik penggunaan normal, karena UI tidak memungkinkan pengguna mengisi dengan format yang salah seperti mengisi field content dengan huruf misalnya. Oleh karena itu, kondisi ini tidak dapat diuji dalam kasus uji ini.	Alur tidak terlewati
3	1-2-3-4-5-6-9	Ketika pengguna mengisi data pada form create akun dengan data yang valid, maka data tersebut akan disimpan ke database dan muncul pesan berhasil menyimpan data	Berhasil	Alur terlewati

### Fungsi Edit data akun

```
public function editAkun($no_reff = null){
    $title = 'Edit';
    $titleTag = 'Edit Data Akun';
    $action = 'data_akun/edit/'.$no_reff;
    $content = 'user/form_akun';

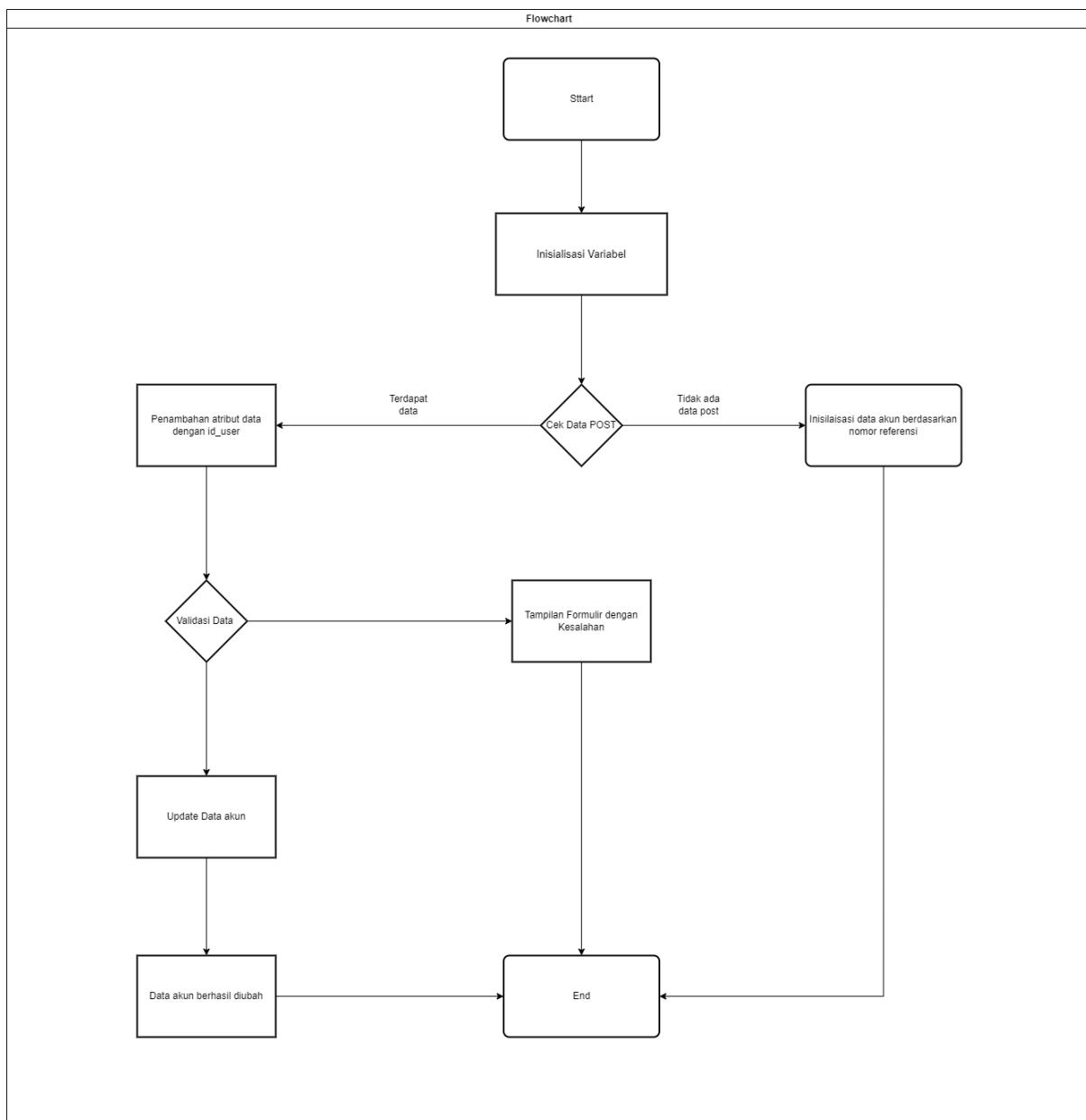
    if(!$_POST){
        $data = (object) $this->akun->getAkunByNo($no_reff);
    }else{
        $data = (object) $this->input->post(null,true);
        $data->id_user = $this->session->userdata('id');
    }
}
```

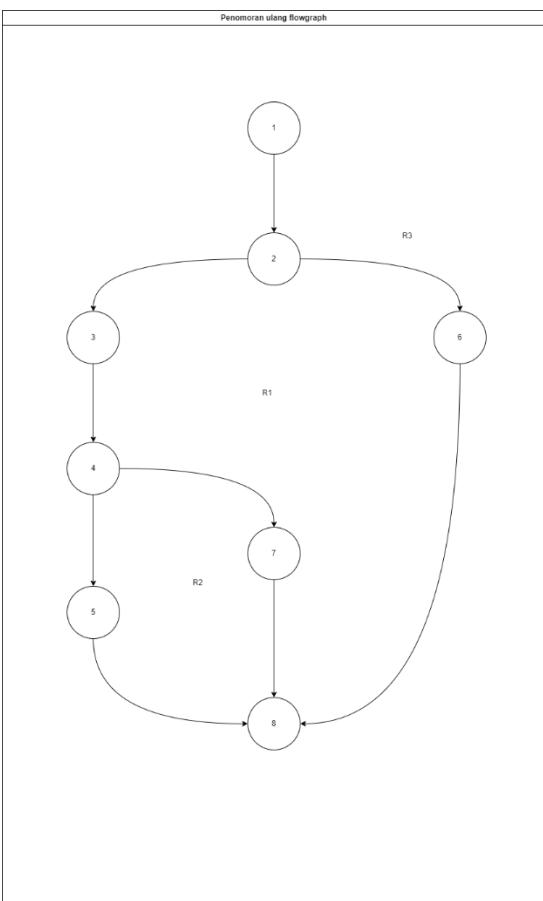
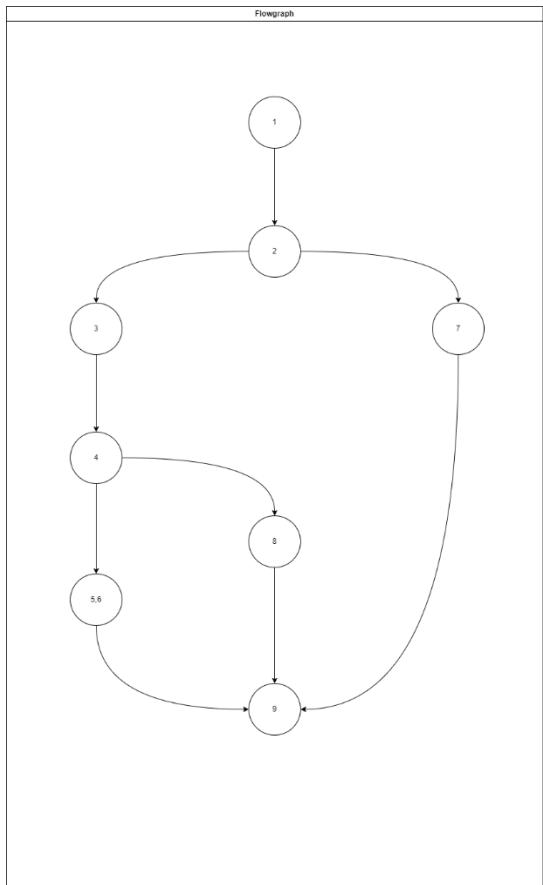
```

if(!$this->akun->validate()){
    $this->load->view('template',compact('content','title','action','data','titleTag'));
    return;
}

$this->akun->updateAkun($no_reff,$data);
$this->session->set_flashdata('berhasil','Data Akun Berhasil Di Ubah');
redirect('data_akun');
}

```





- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 9 - 8 + 2 = 3$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$2 + 1 = 3$$

- Independent path

Path 1 = 1-2-7-9

Path 2 = 1-2-3-4-8-9

Path 3 = 1-2-3-4-5-6-9

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5	6	7	8
1		1						
2			1			1		
3				1				
4					1		1	
5								1
6								1
7								1
8								

- Predicate Node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 2 + 1 = 3$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-7-9	Ketika pengguna ingin mengedit data akun, maka akan dilakukan pencarian terlebih dahulu semisal data akun tidak ada maka proses akan selesai.	Kondisi if(!\$_POST) pada pengecekan apakah terdapat data POST. Jika tidak ada data POST, maka: Objek \$data diinisialisasi dengan data akun berdasarkan nomor referensi menggunakan fungsi getAkunByNo dari objek \$this->akun. Oleh karena itu, kondisi ini tidak dapat diuji dalam kasus uji ini.	Alur tidak terlewati
2	1-2-3-4-8-9	Ketika pengguna mengisi data pada form edit data akun dengan data yang menyebabkan validasi gagal, maka sistem akan menampilkan pesan kesalahan	kondisi if(!\$this->akun->validate()) pada kode tidak dapat terjadi dalam praktik penggunaan normal, karena UI tidak memungkinkan pengguna mengisi dengan format yang salah seperti mengisi field content dengan huruf misalnya. Oleh karena itu, kondisi ini tidak dapat diuji dalam kasus uji ini.	Alur tidak terlewati
3	1-2-3-4-5-6-9	Ketika pengguna mengisi data pada form edit data akun dengan data yang valid, maka data tersebut akan disimpan ke database dan muncul pesan berhasil menubah data.	Berhasil	Alur terlewati

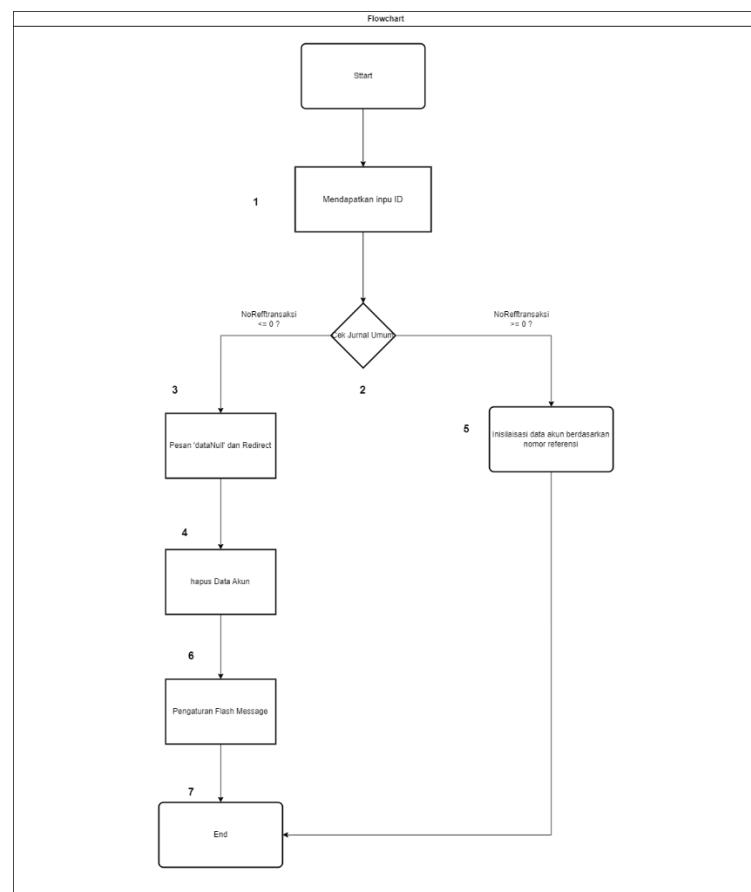
### Fungsi delete akun

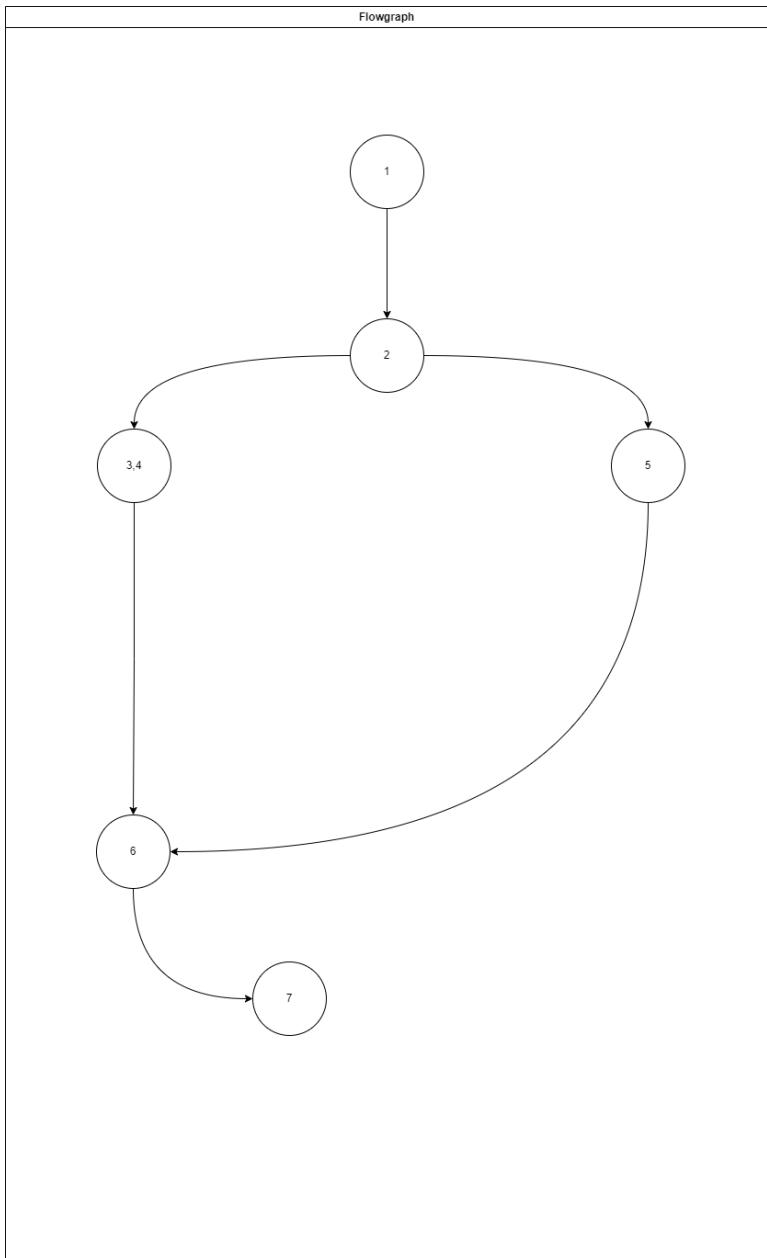
```
public function deleteAkun(){
    $id = $this->input->post('id',true);
    $noReffTransaksi = $this->jurnal->countJurnalNoReff($id);
    if($noReffTransaksi >= 0 ){
        $this->session->set_flashdata('dataNull','No.Reff '.$id.' Tidak
Bisa Di Hapus Karena Data Akun Ada Di Jurnal Umum');
        redirect('data_akun');
    }
}
```

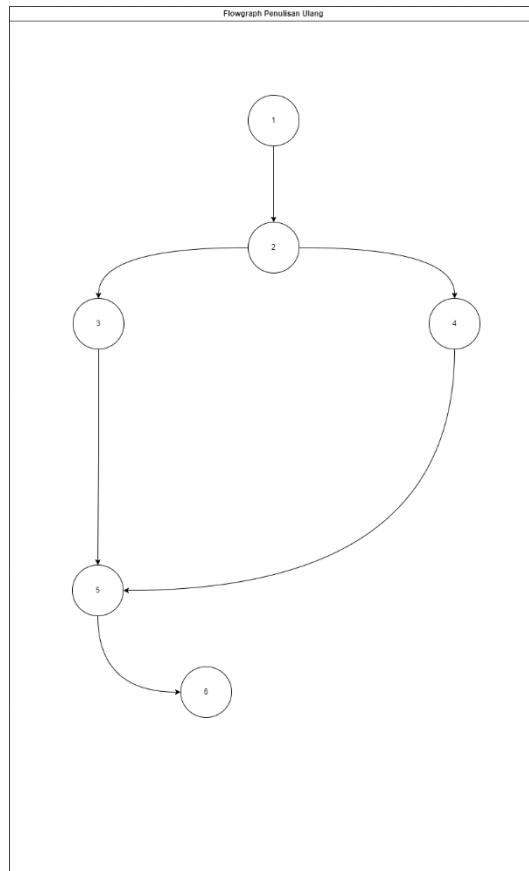
```

    $this->akun->deleteAkun($id);
    $this->session->set_flashdata('berhasilHapus','Data akun dengan
No.Reff '.$id.' berhasil di hapus');
    redirect('data_akun');
}

```







- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 6 - 6 + 2 = 2$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$1 + 1 = 2$$

- Independent path

Path 1 = 1-2-5-6-7

Path 2 = 1-2-3-4-6-7

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5	6
1		1				

2			1	1		
3					1	
4					1	
5						1
6						

- Predicate Node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 1 + 1 = 2$$

- Kesimpulan pengujian white box

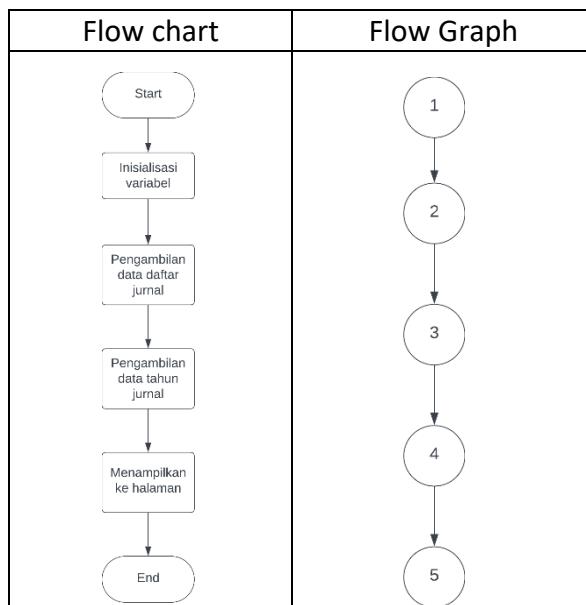
No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-5-6-7	Ketika pengguna ingin menghapus akun, maka akan dilakukan pencarian terlebih dahulu dari nomor id. Jika pencarian melalui nomor referensi maka akan kembali ke data akun.	Kondisi \$noReffTransaksi lebih besar atau sama dengan 0 (ada jurnal umum terkait): Set pesan flash 'dataNull' untuk memberi informasi bahwa data akun tidak dapat dihapus karena masih digunakan dalam jurnal umum. Redirect ke halaman 'data_akun'. Oleh karena itu, kondisi ini tidak dapat diuji dalam kasus uji ini.	Alur tidak terlewati
2	1-2-3-4-6-7	Ketika pengguna ingin menghapus akun, maka akan dilakukan pencarian terlebih dahulu dari nomor id. Jika mendapat notifikasi data null dan redirect akan muncul notifikasi data akun telah berhasil dihapus.	Kondisi \$noReffTransaksi kurang dari 0 (tidak ada jurnal umum terkait): proses akan dilanjutkan untuk penghapusan data akun. Oleh karena itu, kondisi ini dapat diuji dalam kasus uji ini.	Alur terlewati

### C. JURNAL UMUM

Fungsi jurnal umum utama:

- Procedure Sort:

```
public function jurnalUmum(){
    $titleTag = 'Jurnal Umum';
    $content = 'user/jurnal_umum_main';
    $listJurnal = $this->jurnal->getJurnalByYearAndMonth();
    $tahun = $this->jurnal->getJurnalByYear();
    $this->load->view('template', compact('content', 'listJurnal', 'titleTag', 'tahun'));
}
```



- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 4 - 5 + 2 = 1$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

Jumlah region = 0 + 1 = 1

- Independent Path

Path 1 = 1-2-3-4-5

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5
1		1			
2			1		
3				1	
4					1
5					

- Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 0 + 1 = 1$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-3-4-5	Ketika pengguna klik menu Halaman jurnal umum, maka sistem menampilkan daftar jurnal berdasarkan bulan & tahun	Berhasil	Alur terlewati

### Fungsi detail jurnal

- Procedure Sort

```

public function jurnalUmumDetail(){
    $content = 'user/jurnal_umum';
    $titleTag = 'Jurnal Umum';

    $bulan = $this->input->post('bulan',true);
    $tahun = $this->input->post('tahun',true);
    $journals = null;

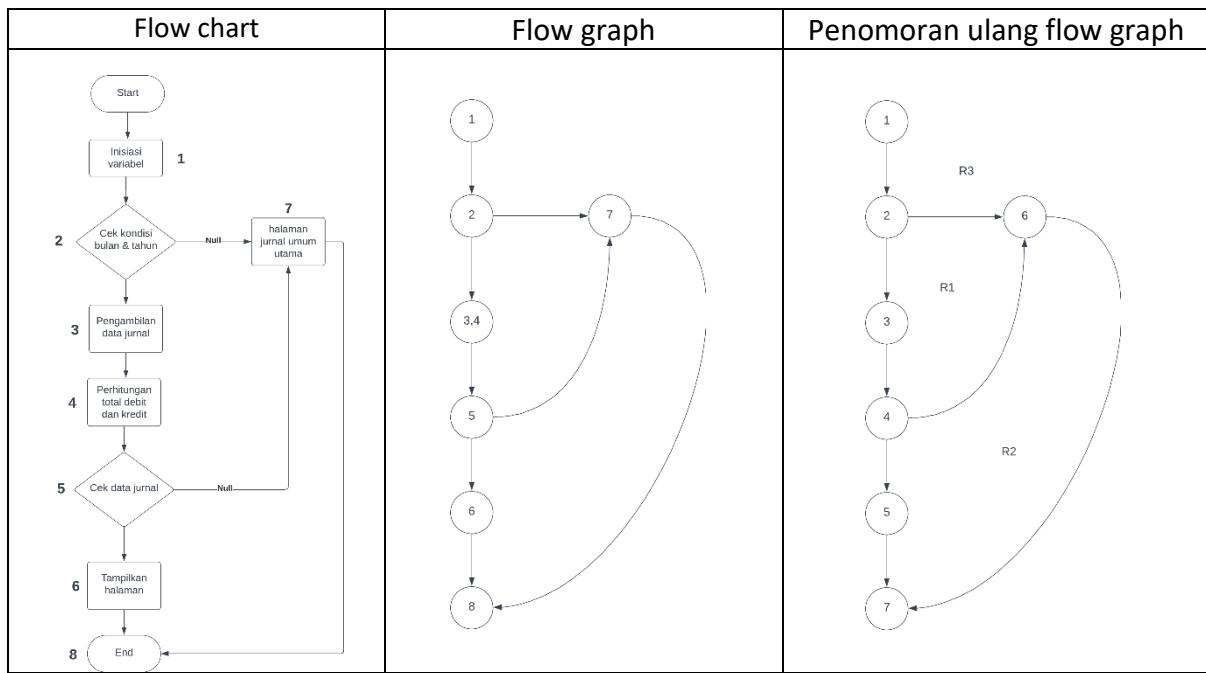
    if(empty($bulan) || empty($tahun)){
        redirect('jurnal_umum');
    }

    $journals = $this->jurnal->getJurnalJoinAkunDetail($bulan,$tahun);
    $totalDebit = $this->jurnal->getTotalSaldoDetail('debit',$bulan,$tahun);
    $totalKredit = $this->jurnal->getTotalSaldoDetail('kredit',$bulan,$tahun);

    if($journals==null){
        $this->session->set_flashdata('dataNull','Data Jurnal Dengan Bulan '.$bulan.' Pada Tahun '.date('Y',strtotime($tahun)).' Tidak Di Temukan');
        redirect('jurnal_umum');
    }

    $this->load->view('template',compact('content','journals','totalDebit','totalKredit','titleTag'));
}

```



- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 8 - 7 + 2 = 3$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$2 + 1 = 3$$

- Independent path

Path 1 = 1-2-7-8

Path 2 = 1-2-3-4-5-7-8

Path 3 = 1-2-3-4-5-6-8

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5	6	7
1		1					
2			1			1	
3				1			
4					1	1	
5							1
6							1
7							

- Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

```

public function createJurnal(){
    $title = 'Tambah';
    $content = 'user/form_jurnal';
    $action = 'jurnal_umum/tambah';
    $tgl_input = date('Y-m-d H:i:s');
    $id_user = $this->session->userdata('id');
    $titleTag = 'Tambah Jurnal Umum';

    if(!$_POST){
        $data = (object) $this->jurnal->getDefaultValue();
    }else{
        $data = (object) [
            'id_user'=>$id_user,
            'no_reff'=>$this->input->post('no_reff',true),
            'tgl_input'=>$tgl_input,
            'tgl_transaksi'=>$this->input->post('tgl_transaksi',true),
            'jenis_saldo'=>$this->input->post('jenis_saldo',true),
            'saldo'=>$this->input->post('saldo',true)
        ];
    }

    if(!$this->jurnal->validate()){
        $this->load->view('template',compact('content','title','action','data','titleTag'));
        return;
    }

    $this->jurnal->insertJurnal($data);
    $this->session->set_flashdata('berhasil','Data Jurnal Berhasil Di Tambahkan');
    redirect('jurnal_umum');
}

```

$$V(G) = 2 + 1 = 3$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-7-8	Jika bulan dan tahun kosong, maka kembali ke halaman jurnal umum utama	Pengujian tidak dapat terjadi, karena dalam tampilan sistem bulan dan tahun pada fitur pencarian sudah memiliki nilai default atau tidak dapat dipilih kosong oleh pengguna. Oleh karena itu, kondisi ini tidak teruji dalam kasus uji ini.	Alur tidak terlewati
2	1-2-3-4-5-7-8	Jika data jurnal pada bulan dan tahun yang dipilih tidak ada, tampil pesan data tidak ditemukan	Berhasil	Alur terlewati
3	1-2-3-4-5-6-8	Jika data jurnal pada bulan dan tahun yang dipilih ada, maka sistem menampilkan data jurnal pada bulan dan tahun tersebut	Berhasil	Alur terlewati

- **Fungsi tambah jurnal**
- Procedure sort

Flowchart	Flowgraph	Penomoran ulang flowgraph
<pre> graph TD     Start([Start]) --&gt; 1[Inisialisasi variabel]     1 --&gt; 2{Pengecekan data}     2 --&gt; 3[Nilai inputan]     3 --&gt; 4{Validasi data}     4 --&gt; 5[Simpan data ke database]     5 --&gt; 6[Menampilkan pesan berhasil]     6 --&gt; End([End])     2 -- Null --&gt; 7[Nilai default]     7 --&gt; 9([End])     4 -- Salah --&gt; 8[Pesan kesalahan]     8 --&gt; 9   </pre>		

- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 9 - 8 + 2 = 3$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$2 + 1 = 3$$

- Independent path

$$\text{Path 1} = 1-2-7-9$$

$$\text{Path 2} = 1-2-3-4-8-9$$

$$\text{Path 3} = 1-2-3-4-5-6-9$$

- Pembentukan matriks dan perhitungan  $V(G)$

	1	2	3	4	5	6	7	8
1		1						
2			1			1		
3				1				
4					1		1	
5								1
6								1
7								1
8								

- Predicate Node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 2 + 1 = 3$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-7-9	Ketika pengguna tidak mengisi data pada form tambah jurnal, maka data jurnal akan diisi dengan data default	Kondisi <code>if(!\$_POST)</code> pada kode tidak dapat terjadi dalam praktik penggunaan normal karena UI tidak memungkinkan pengguna mengirimkan formulir tanpa mengisi data yang diperlukan. Oleh karena itu, kondisi ini tidak dapat diuji dalam kasus uji ini.	Alur tidak terlewati

2	1-2-3-4-8-9	Ketika pengguna mengisi data pada form tambah jurnal dengan data yang menyebabkan validasi gagal, maka sistem akan menampilkan pesan kesalahan	kondisi if(!\$this->jurnal->validate()) pada kode tidak dapat terjadi dalam praktik penggunaan normal, karena UI tidak memungkinkan pengguna mengisi dengan format yang salah seperti mengisi field saldo dengan huruf misalnya. Oleh karena itu, kondisi ini tidak dapat diuji dalam kasus uji ini.	Alur tidak terlewati
3	1-2-3-4-5-6-9	Ketika pengguna mengisi data pada form tambah jurnal dengan data yang valid, maka data tersebut akan disimpan ke database dan muncul pesan berhasil menyimpan data	Berhasil	Alur terlewati

### Fungsi edit jurnal

- Procedure Sort

```

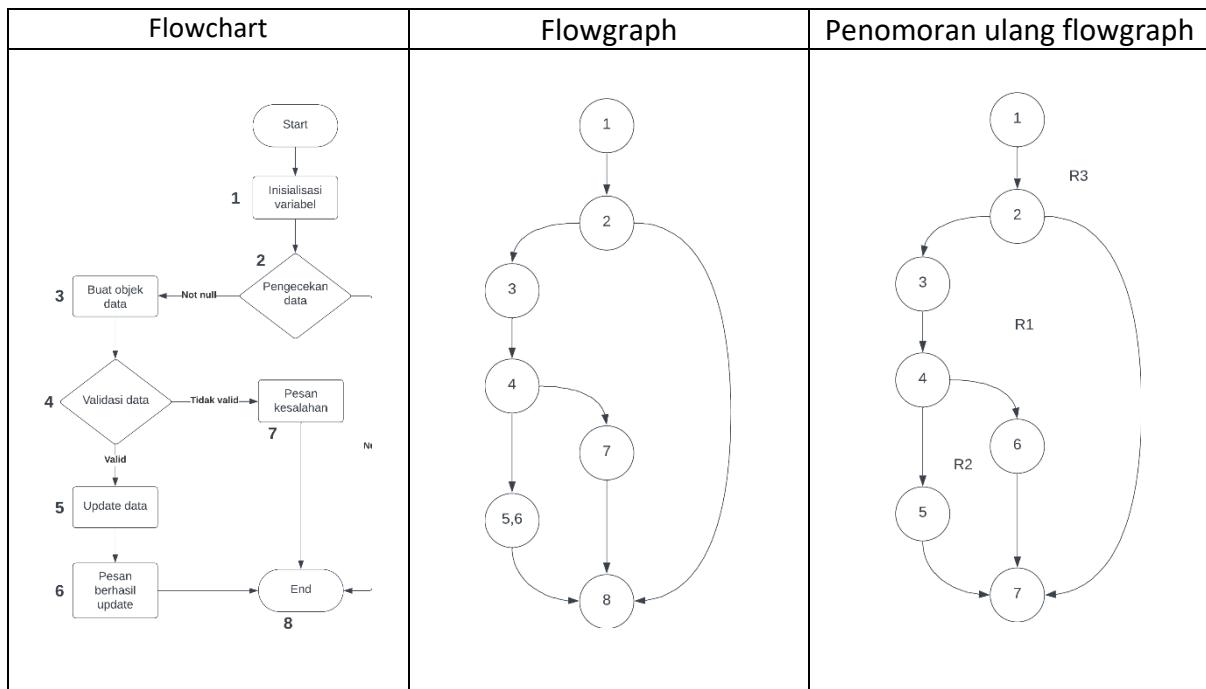
public function editJurnal()
{
    $title = 'Edit';
    $content = 'user/form_jurnal';
    $action = 'jurnal_umum/edit';
    $tgl_input = date('Y-m-d H:i:s');
    $id_user = $this->session->userdata('id');
    $titleTag = 'Edit Jurnal Umum';

    if ($_POST) {
        $data = (object) [
            'id_user' => $id_user,
            'no_reff' => $this->input->post('no_reff', true),
            'tgl_input' => $tgl_input,
            'tgl_transaksi' => $this->input->post('tgl_transaksi', true),
            'jenis_saldo' => $this->input->post('jenis_saldo', true),
            'saldo' => $this->input->post('saldo', true)
        ];
        $id = $this->input->post('id', true);
    }

    if (!$this->jurnal->validate()) {
        $this->load->view('template', compact('content', 'title', 'action', 'data', 'id', 'titleTag'));
        return;
    }

    $this->jurnal->updateJurnal($id, $data);
    $this->session->set_flashdata('berhasil', 'Data Jurnal Berhasil Di Ubah');
    redirect('jurnal_umum');
}

```



- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 8 - 7 + 2 = 3$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$2 + 1 = 3$$

- Independent path

Path 1 = 1-2-8

Path 2 = 1-2-3-4-7-8

Path 3 = 1-2-3-4-5-6-8

- Pembentukan dan perhitungan matriks V(G)

	1	2	3	4	5	6	7
1		1					
2			1				1
3				1			
4					1	1	
5							1
6							1
7							

- Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 2 + 1 = 3$$

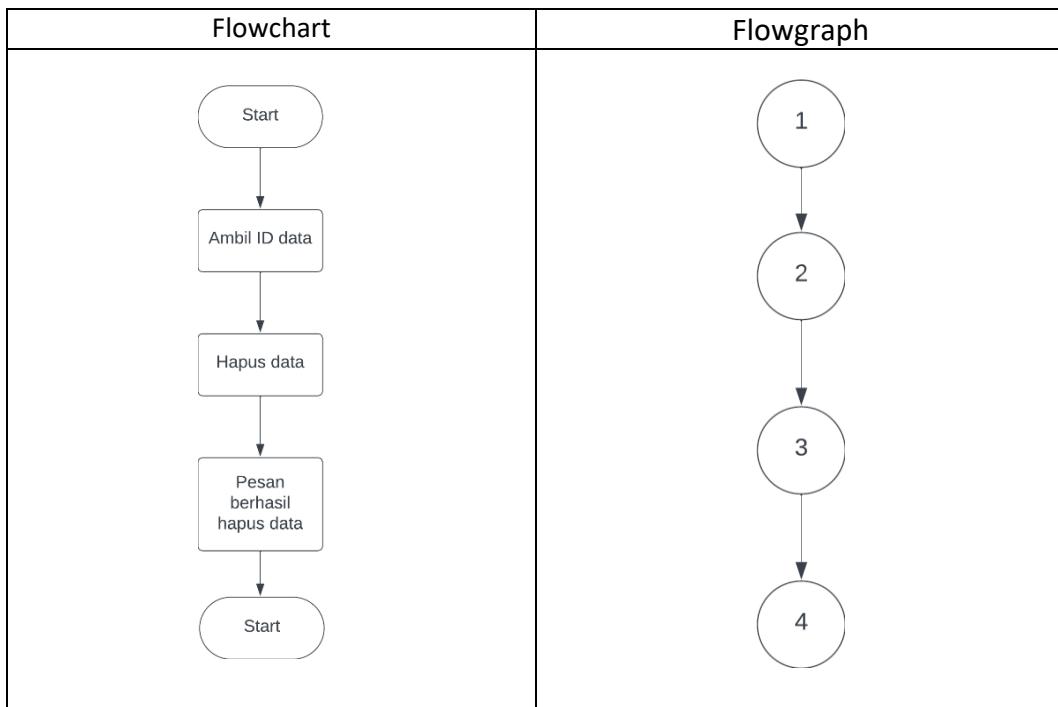
- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-8	Ketika tidak mengeklik tombol edit setelah mengisi form edit, maka update ada tidak akan terjadi	Berhasil	Alur terlewati
2	1-2-3-4-7-8	Ketika pengguna mengisi data pada form tambah jurnal dengan data yang menyebabkan validasi gagal, maka sistem akan menampilkan pesan kesalahan	kondisi if(!\$this->jurnal->validate()) pada kode tidak dapat terjadi dalam praktik penggunaan normal, karena UI tidak memungkinkan pengguna mengisi dengan format yang salah seperti mengisi field saldo dengan huruf misalnya. Oleh karena itu, kondisi ini tidak dapat diuji dalam kasus uji ini.	Alur tidak terlewati
3	1-2-3-4-5-6-8	Ketika pengguna mengisi data pada form edit jurnal dengan data yang valid, maka data tersebut akan disimpan ke database dan muncul pesan berhasil update data	Berhasil	Alur terlewati

### Fungsi hapus jurnal

- Procedure sort

```
public function deleteJurnal()
{
    $id = $this->input->post('id', true);
    $this->jurnal->deleteJurnal($id);
    $this->session->set_flashdata('berhasilHapus', 'Data Jurnal berhasil di hapus');
    redirect('jurnal_umum');
}
```



- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 3 - 4 + 2 = 1$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$0 + 1 = 1$$

- Independent path

Path 1 = 1-2-3-4

- Pembentukan dan perhitungan matriks V(G)

	1	2	3	4
1		1		
2			1	
3				1
4				

- Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 0 + 1 = 1$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-3-4	Ketika pengguna klik salah satu data yang akan dihapus, maka data tersebut akan terhapus dari database	Berhasil	Alur terlewati

## D. PENGUJIAN WHITE BOX PADA MENU BUKU BESAR

### Fungsi buku besar utama

- Procedure sort

```
public function bukuBesarDetail()
{
    $content = 'user/buku_besar';
    $titleTag = 'Buku Besar';

    $bulan = $this->input->post('bulan', true);
    $tahun = $this->input->post('tahun', true);

    if (empty($bulan) || empty($tahun)) [
        redirect('buku_besar');
    ]

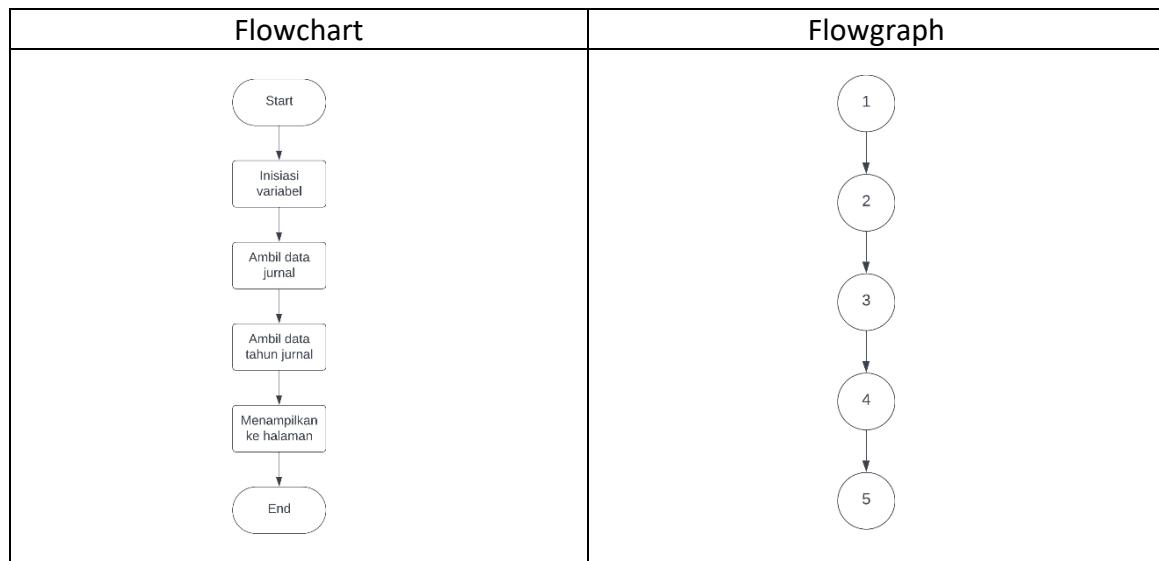
    $dataAkun = $this->akun->getAkunByMonthYear($bulan, $tahun);
    $data = null;
    $saldo = null;

    foreach ($dataAkun as $row) {
        $data[] = (array) $this->jurnal->getJurnalByNoReffMonthYear($row->no_reff, $bulan, $tahun);
        $saldo[] = (array) $this->jurnal->getJurnalByNoReffSaldoMonthYear($row->no_reff, $bulan, $tahun);
    }

    if ($data == null || $saldo == null) {
        $this->session->set_flashdata('dataNull', 'Data Buku Besar Dengan Bulan ' . bulan($bulan) . ' Pada Tahun ' . date('Y', strtotime($tahun)) . ' Tidak Di Temukan');
        redirect('buku_besar');
    }

    $jumlah = count($data);

    $this->load->view('template', compact('content', 'titleTag', 'dataAkun', 'data', 'jumlah', 'saldo'));
}
```



- Cyclomatic Complexity

$$V(G) = E - N + 2$$

$$V(G) = 4 - 5 + 2 = 1$$

- Region testing

Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)

$$0 + 1 = 1$$

- Independent path  
Path 1 = 1-2-3-4-5

- Pembentukan dan perhitungan matriks V(G)

	1	2	3	4	5
1		1			
2			1		
3				1	
4					1
5					

- Predicate node

$V(G) = \text{Jumlah predicate node} + 1$

$$V(G) = 0 + 1 = 1$$

- Kesimpulan pengujian white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-3-4-5	Ketika pengguna klik halaman buku besar, maka tampil halaman utama buku besar	Berhasil	Alur terlewati

## Fungsi Detail buku besar

- Procedure sort

```

public function bukuBesarDetail()
{
    $content = 'user/buku_besar';
    $titleTag = 'Buku Besar';

    $bulan = $this->input->post('bulan', true);
    $tahun = $this->input->post('tahun', true);

    if (empty($bulan) || empty($tahun)) {
        redirect('buku_besar');
    }

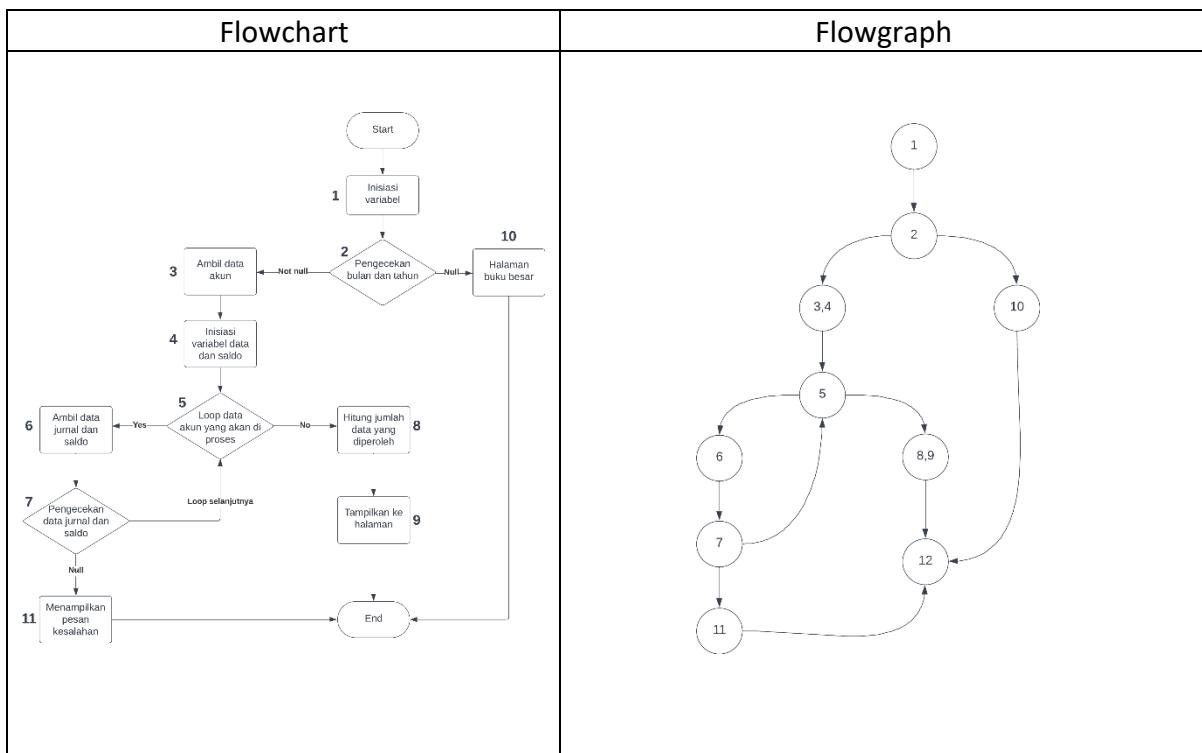
    $dataAkun = $this->akun->getAkunByMonthYear($bulan, $tahun);
    $data = null;
    $saldo = null;

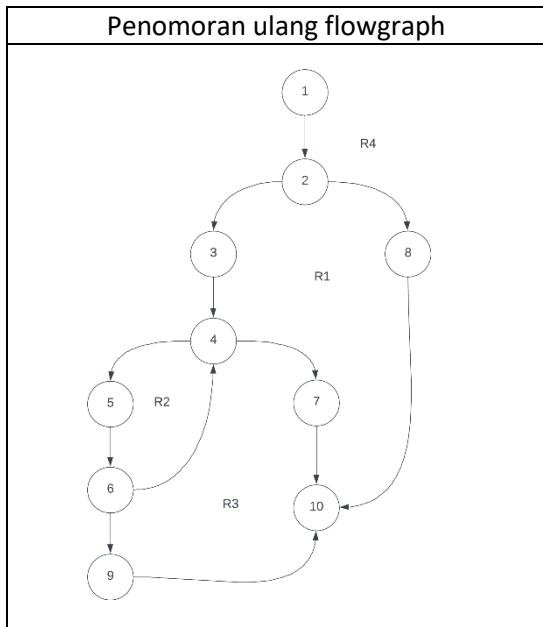
    foreach ($dataAkun as $row) {
        $data[] = (array) $this->jurnal->getJurnalByNoReffMonthYear($row->no_reff, $bulan, $tahun);
        $saldo[] = (array) $this->jurnal->getJurnalByNoReffSaldoMonthYear($row->no_reff, $bulan, $tahun);
    }

    if ($data == null || $saldo == null) {
        $this->session->set_flashdata('dataNull', 'Data Buku Besar Dengan Bulan ' . bulan($bulan) . ' Pada Tahun ' . date('Y', strtotime($tahun)) . ' Tidak Di Temukan');
        redirect('buku_besar');
    }

    $jumlah = count($data);
    $this->load->view('template', compact('content', 'titleTag', 'dataAkun', 'data', 'jumlah', 'saldo'));
}

```





- Cyclomatic Complexity  
 $V(G) = E - N + 2$   
 $V(G) = 12 - 10 + 2 = 4$
  - Region testing  
Jumlah Region = Jumlah kurva tertutup + 1 (region terluar dalam flowgraph)  
 $3 + 1 = 4$
  - Independent path  
Path 1 = 1-2-8-10  
Path 2 = 1-2-3-4-5-6-9-10  
Path 3 = 1-2-3-4-5-6-4-7-10
  - Pembentukan dan perhitungan matriks  $V(G)$

- Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 3 + 1 = 4$$

- Kesimpulan perhitungan white box

No	Kasus Uji	Hasil yang diharapkan	Hasil sesuai uji kasus	Keterangan
1	1-2-8-10	Ketika data bulan dan tahun yang dipilih tidak ada, maka kembali ke halaman buku besar utama	Pengujian tidak dapat terjadi, karena dalam tampilan sistem bulan dan tahun pada fitur pencarian sudah memiliki nilai default atau tidak dapat dipilih kosong oleh pengguna. Oleh karena itu, kondisi ini tidak teruji dalam kasus uji ini.	Alur tidak terlewati
2	1-2-3-4-5-6-9-10	Ketika data buku besar pada bulan dan tahun yang dipilih tidak ada, maka muncul pesan kesalahan	Berhasil	Alur terlewati
3	1-2-3-4-5-6-4-7-10	Ketika data buku besar pada bulan dan tahun yang dipilih ada, maka sistem menampilkan data buku besar pada bulan dan tahun tersebut	Berhasil	Alur terlewati

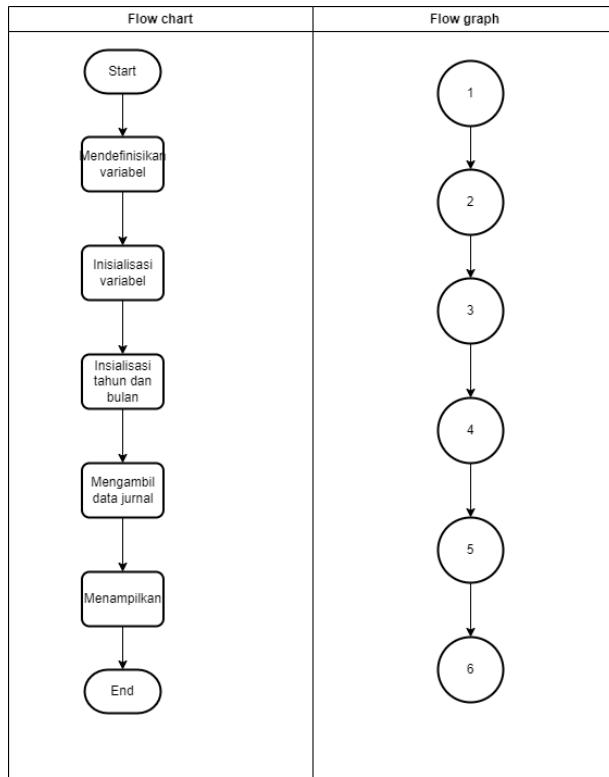
## E. Pengujian White Box Pada Neraca Saldo

### ➤ Neraca Saldo

```
public function neracaSaldo(){
    $titleTag = 'Neraca Saldo';
    $content = 'user/neraca_saldo_main';
    $listJurnal = $this->jurnal->getJurnalByYearAndMonth();
    $tahun = $this->jurnal->getJurnalByYear();
    $this->load->view('template',compact('content','listJurnal','titleTag','tahun'));
}
```

#### 1. Flowchart dan Flowgraph

Flowchart ini menggambarkan proses menampilkan neraca saldo dari awal hingga akhir. Flowchart ini dapat digunakan untuk memahami proses menampilkan neraca saldo. Selain itu, flowchart ini juga dapat digunakan untuk mengonversikan proses menampilkan neraca saldo ke dalam bentuk flowgraph.



Setelah membuat flowchart, langkah selanjutnya adalah mengonversinya ke dalam bentuk flowgraph. Tujuan dibuat flowgraph sendiri adalah untuk menghitung cyclomatic complexity. Cyclomatic complexity adalah ukuran kompleksitas algoritma yang dihitung berdasarkan jumlah node dan jumlah edge dari suatu flowgraph.

## 2. Cyclomatic Complexity

Untuk menghitung menggunakan formula  $V(G) = E - N + 2$ . Berdasarkan penerapan flowgraph maka dapat diketahui nilainya sebagai berikut:

$$E: 5$$

$$N: 6$$

Masukan ke dalam formula sebagai berikut:

$$V(G) = E - N + 2$$

$$V(G) = 5 - 6 + 2$$

$$V(G) = 1$$

## 3. Region Testing

Jumlah Region = Jumlah kurva tertutup + 1(region terluar)

Jumlah Region =  $0 + 1 = 1$

#### 4. Independen Path

Dari perhitungan cyclomatic didapatkan 1 jalur independent yaitu

Path 1 = 1-2-3-4-5-6

#### 5. Matriks

	1	2	3	4	5	6
1		1				
2			1			
3				1		
4					1	
5						1
6						

#### 6. Predicate node

$V(G) = \text{Jumlah predicate node} + 1$

$V(G) = 0 + 1 = 1$

#### 7. Test Case

No	Kasus Uji	Hasil yang diharapkan	Hasil	Keterangan
1	1-2-3-4-5-6	Ketika pengguna klik menu Halaman neraca saldo, maka sistem menampilkan daftar neraca berdasarkan bulan & tahun	sistem menampilkan daftar neraca berdasarkan bulan & tahun	Valid Alur terlewati

#### ➤ Detail Neraca Saldo

```
public function neracaSaldoDetail(){
    $content = 'user/neraca_saldo';
    $titleTag = 'Neraca Saldo';

    $bulan = $this->input->post('bulan', true);
```

```

$tahun = $this->input->post('tahun',true);

if(empty($bulan) || empty($tahun)){
    redirect('neraca_saldo');
}

$dataAkun = $this->akun->getAkunByMonthYear($bulan,$tahun);
$data = null;
$saldo = null;

foreach($dataAkun as $row){
    $data[] = (array) $this->jurnal->getJurnalByNoReffMonthYear($row->no_reff,$bulan,$tahun);
    $saldo[] = (array) $this->jurnal-
>getJurnalByNoReffSaldoMonthYear($row->no_reff,$bulan,$tahun);
}

if($data == null || $saldo == null){
    $this->session->set_flashdata('dataNull','Neraca Saldo Dengan
Bulan '.$bulan.' Pada Tahun '.date('Y',strtotime($tahun)).' Tidak Di
Temukan');
    redirect('neraca_saldo');
}

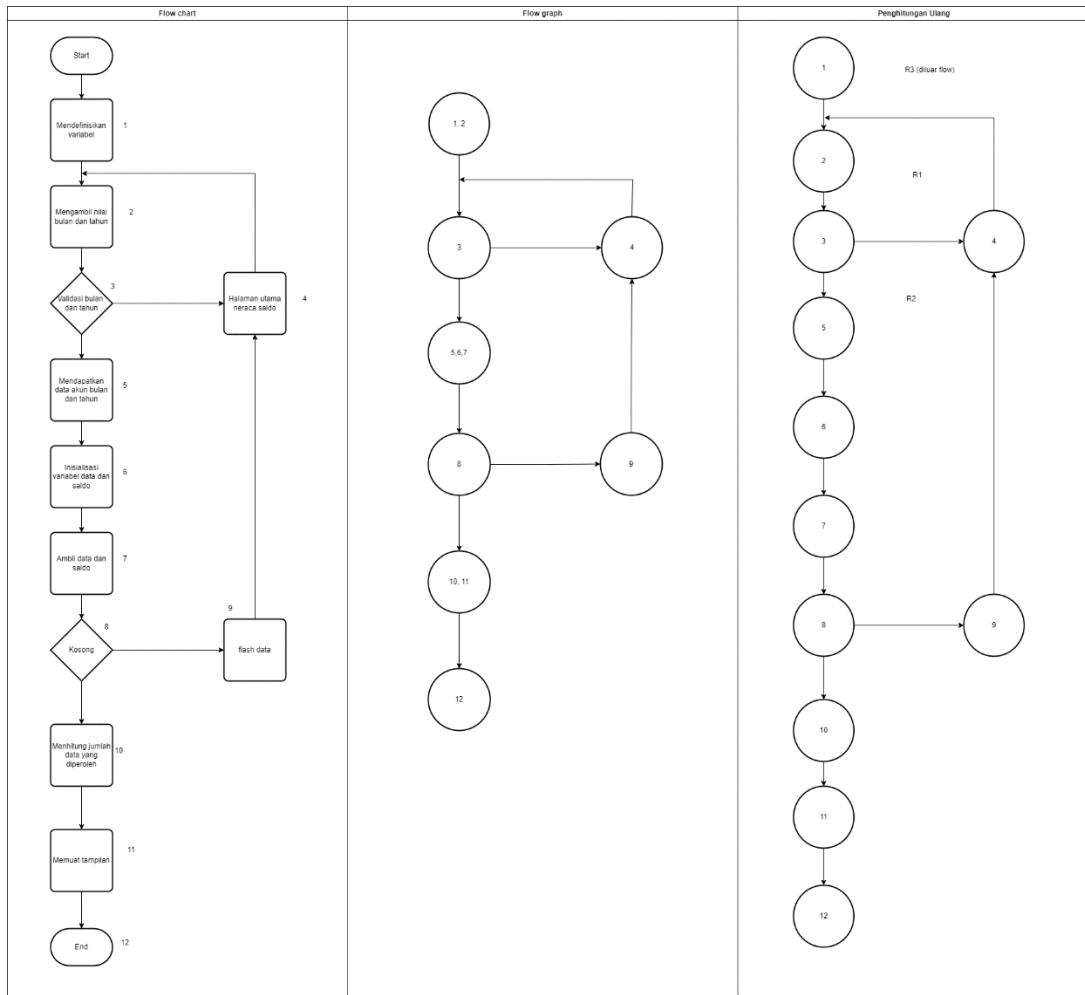
$jumlah = count($data);

$this->load-
>view('template',compact('content','titleTag','dataAkun','data','jumlah','saldo'));
}

```

## 1. Flowchart dan Flowgraph

Flowchart ini menggambarkan proses menampilkan detail neraca saldo dari awal hingga akhir. Flowchart ini dapat digunakan untuk memahami proses menampilkan detail neraca saldo. Selain itu, flowchart ini juga dapat digunakan untuk mengonversikan proses menampilkan neraca saldo ke dalam bentuk flowgraph.



Setelah membuat flowchart, langkah selanjutnya adalah mengonversinya ke dalam bentuk flowgraph. Tujuan dibuat flowgraph sendiri adalah untuk menghitung cyclomatic complexity. Cyclomatic complexity adalah ukuran kompleksitas algoritma yang dihitung berdasarkan jumlah node dan jumlah edge dari suatu flowgraph.

## 2. Cyclomatic Complexity

Untuk menghitung menggunakan formula  $V(G) = E - N + 2$ . Berdasarkan penerapan flowgraph maka dapat diketahui nilainya sebagai berikut:

$$E: 9$$

$$N: 8$$

Masukan ke dalam formula sebagai berikut:

$$V(G) = E - N + 2$$

$$V(G) = 9 - 8 + 2$$

$$V(G) = 3$$

### 3. Region Testing

Jumlah Region = Jumlah kurva tertutup + 1(region terluar)

$$\text{Jumlah Region} = 2 + 1 = 3$$

### 4. Independen Path

Dari perhitungan cyclomatic didapatkan 1 jalur independent yaitu

$$\text{Path 1} = 1-2-3-5-6-7-8-10-11-12$$

$$\text{Path 2} = 1-2-3-4$$

$$\text{Path 3} = 1-2-3-5-6-7-8-9-4$$

### 5. Matriks

	1	2	3	4	5	6	7	8	9	10	11	12
1		1										
2			1									
3				1	1							
4		1										
5						1						
6							1					
7								1				
8									1	1		
9				1								
10											1	
11												1
12												

### 6. Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 2 + 1 = 3$$

### 7. Case Test

No	Kasus Uji	Hasil yang diharapkan	Hasil	Keterangan
1	1-2-3-5-6-7-8-10-11-12	Ketika pengguna memilih bulan dan tahun, maka sistem menampilkan detail data neraca saldo berdasarkan bulan & tahun	system menampilkan detail data neraca saldo berdasarkan bulan & tahun	Valid Alur terlewati
2	1-2-3-4	Jika bulan atau tahun kosong, system mengarahkan pengguna kembali ke halaman utama neraca saldo.	system mengarahkan pengguna kembali ke halaman utama neraca saldo.	Valid Alur terlewati
3	1-2-3-5-6-7-8-9-4	Jika data maupun saldo kosong, akan tampil pesan flash data dan mengarahkan pengguna kembali ke halaman utama neraca saldo.	Tampil pesan flash dan Pengguna diarahkan Kembali ke halaman utama neraca saldo	Valid Alur terlewati

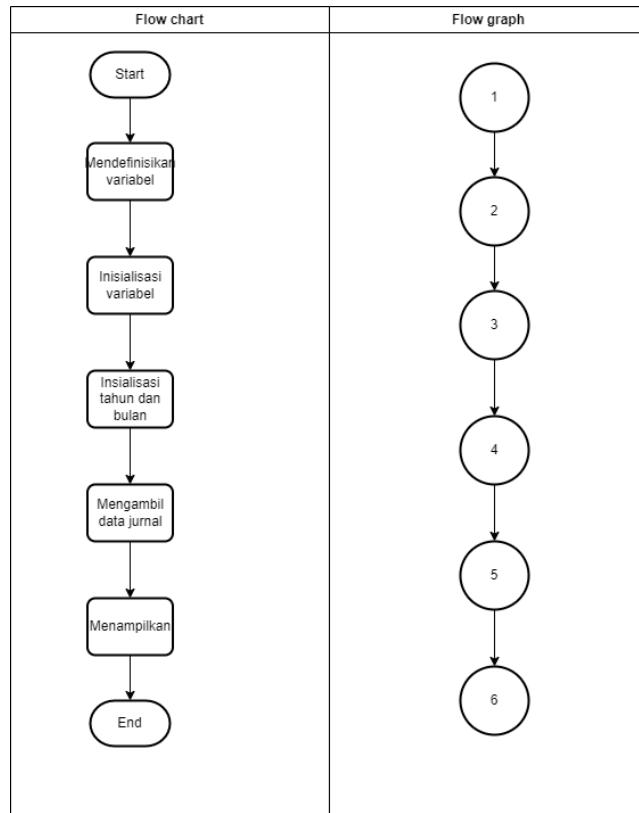
## F. Pengujian White Box Pada Laporan

### ➤ Laporan

```
public function laporan(){
    $titleTag = 'Laporan';
    $content = 'user/laporan_main';
    $listJurnal = $this->jurnal->getJurnalByYearAndMonth();
    $tahun = $this->jurnal->getJurnalByYear();
    $this->load-
>view('template',compact('content','listJurnal','titleTag','tahun'));
}
```

#### 1. Flowchart dan Flowgraph

Flowchart ini menggambarkan proses menampilkan menu laporan dari awal hingga akhir. Flowchart ini dapat digunakan untuk memahami proses menampilkan laporan. Selain itu, flowchart ini juga dapat digunakan untuk mengonversikan proses menampilkan laporan ke dalam bentuk flowgraph.



Setelah membuat flowchart, langkah selanjutnya adalah mengonversinya ke dalam bentuk flowgraph. Tujuan dibuat flowgraph sendiri adalah untuk menghitung cyclomatic complexity. Cyclomatic complexity adalah ukuran kompleksitas algoritma yang dihitung berdasarkan jumlah node dan jumlah edge dari suatu flowgraph.

## 2. Cyclomatic Complexity

Untuk menghitung menggunakan formula  $V(G) = E - N + 2$ . Berdasarkan penerapan flowgraph maka dapat diketahui nilainya sebagai berikut:

$$E: 5$$

$$N: 6$$

Masukan ke dalam formula sebagai berikut:

$$V(G) = E - N + 2$$

$$V(G) = 5 - 6 + 2$$

$$V(G) = 1$$

## 3. Region Testing

Jumlah Region = Jumlah kurva tertutup + 1(region terluar)

Jumlah Region = 0 + 1 = 1

#### 4. Independen Path

Dari perhitungan cyclomatic didapatkan 1 jalur independent yaitu

Path 1 = 1-2-3-4-5-6

#### 5. Matriks

	1	2	3	4	5	6
1		1				
2			1			
3				1		
4					1	
5						1
6						

#### 6. Predicate node

$V(G) = \text{Jumlah predicate node} + 1$

$V(G) = 0 + 1 = 1$

#### 7. Test Case

No	Kasus Uji	Hasil yang diharapkan	Hasil	Keterangan
1	1-2-3-4-5-6	Ketika pengguna klik menu Halaman laporan, maka sistem menampilkan daftar laporan berdasarkan bulan & tahun	sistem menampilkan daftar laporan berdasarkan bulan & tahun	Valid Alur terlewati

#### ➤ Detail Laporan

```
public function laporanCetak(){
    $bulan = $this->input->post('bulan',true);
    $tahun = $this->input->post('tahun',true);
```

```

$titleTag = 'Laporan '.bulan($bulan).' '.$tahun;

$dataAkun = $this->akun->getAkunByMonthYear($bulan,$tahun);

$jurnals = $this->jurnal->getJurnalJoinAkunDetail($bulan,$tahun);
$totalDebit = $this->jurnal-
>getTotalSaldoDetail('debit',$bulan,$tahun);
$totalKredit = $this->jurnal-
>getTotalSaldoDetail('kredit',$bulan,$tahun);

$data = null;
$saldo = null;
foreach($dataAkun as $row){
    $data[] = (array) $this->jurnal->getJurnalByNoReffMonthYear($row-
>no_reff,$bulan,$tahun);
    $saldo[] = (array) $this->jurnal-
>getJurnalByNoReffSaldoMonthYear($row->no_reff,$bulan,$tahun);
}

if($data == null || $saldo == null){
    $this->session->set_flashdata('dataNull','Laporan Dengan Bulan
'.bulan($bulan).' Pada Tahun '.date('Y',strtotime($tahun)).' Tidak Di
Temukan');
    redirect('laporan');
}

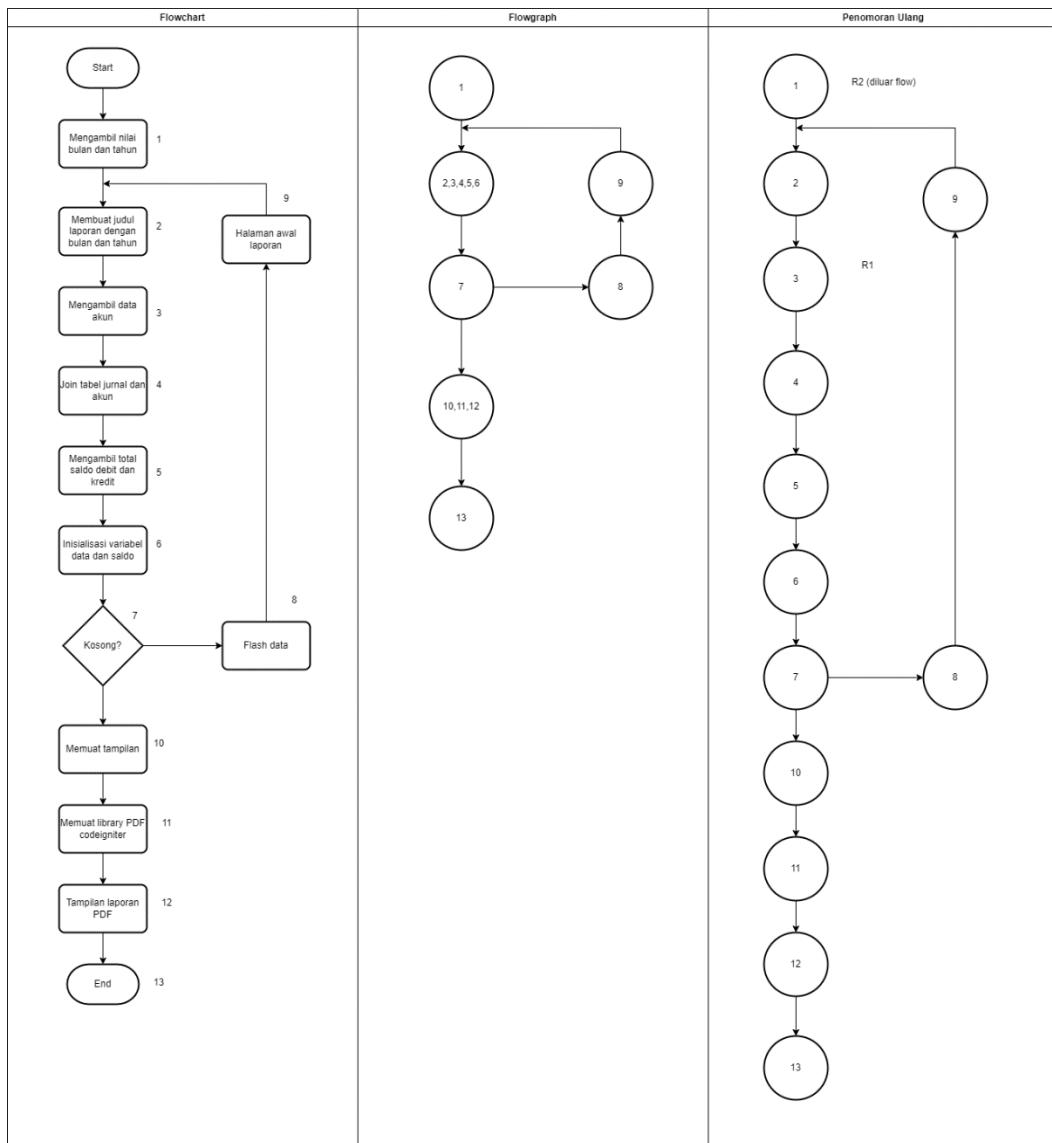
$jumlah = count($data);

$data = $this->load-
>view('user/laporan',compact('titleTag','dataAkun','bulan','tahun','jurnals','
totalDebit','totalKredit','data','saldo','jumlah'),true);
// echo $data;
// die();
$this->load->library('pdf');
$this->pdf->setPaper('A4', 'landscape');
$this->pdf->filename = "laporan_".bulan($bulan).'_'.$tahun;
$this->pdf->load_view('user/laporan', $data);
}

```

## 1. Flowchart dan flowgraph

Flowchart ini menggambarkan proses menampilkan detail cetak laporan dari awal hingga akhir. Flowchart ini dapat digunakan untuk memahami proses menampilkan cetak laporan. Selain itu, flowchart ini juga dapat digunakan untuk mengonversikan proses menampilkan laporan ke dalam bentuk flowgraph.



Setelah membuat flowchart, langkah selanjutnya adalah mengonversinya ke dalam bentuk flowgraph. Tujuan dibuat flowgraph sendiri adalah untuk menghitung cyclomatic complexity. Cyclomatic complexity adalah ukuran kompleksitas algoritma yang dihitung berdasarkan jumlah node dan jumlah edge dari suatu flowgraph.

## 8. Cyclomatic Complexity

Untuk menghitung menggunakan formula  $V(G) = E - N + 2$ . Berdasarkan penerapan flowgraph maka dapat diketahui nilainya sebagai berikut:

$$E: 7$$

$$N: 7$$

Masukan ke dalam formula sebagai berikut:

$$V(G) = E - N + 2$$

$$V(G) = 7 - 7 + 2$$

$$V(G) = 2$$

## 9. Region Testing

Jumlah Region = Jumlah kurva tertutup + 1(region terluar)

Jumlah Region = 1 + 1 = 2

## 10. Independen Path

Dari perhitungan cyclomatic didapatkan 1 jalur independent yaitu

Path 1 = 1-2-3-4-5-6-7-10-11-12-13

Path 2 = 1-2-3-4-5-6-7-8-9

## 11. Matriks

	1	2	3	4	5	6	7	8	9	10	11	12	13
1		1											
2			1										
3				1									
4					1								
5						1							
6							1						
7								1		1			
8									1				
9		1											
10										1			
11											1		
12												1	
13													

## 12. Predicate node

$$V(G) = \text{Jumlah predicate node} + 1$$

$$V(G) = 1 + 1 = 2$$

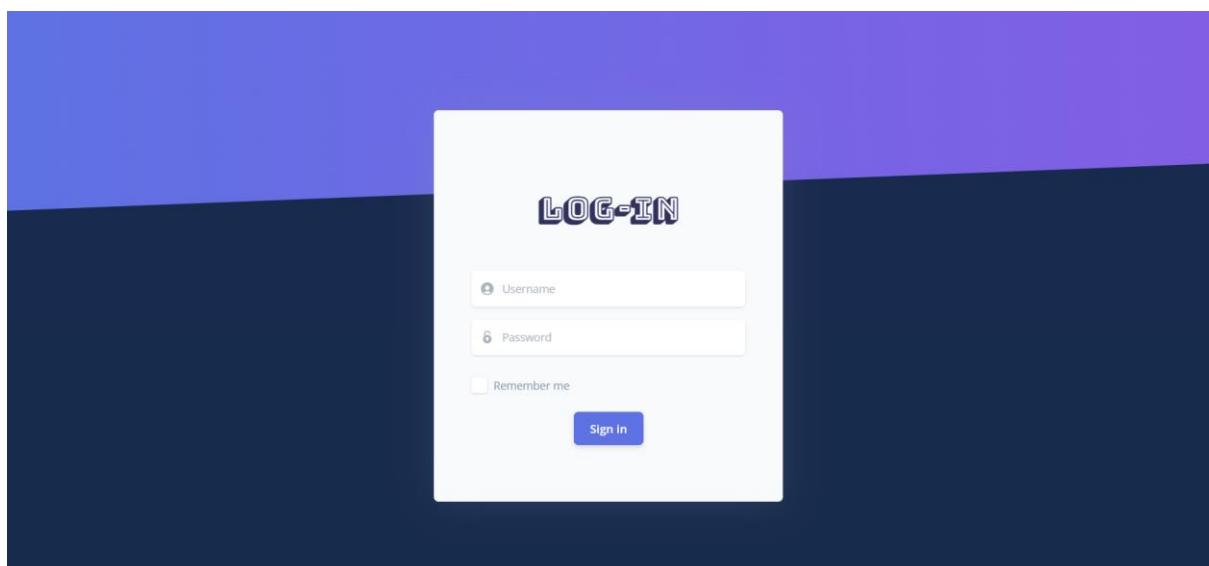
### 13. Case Test

No	Kasus Uji	Hasil yang diharapkan	Hasil	Keterangan
1	1-2-3-4-5-6-7-10-11-12-13	Ketika pengguna memilih bulan dan tahun, maka sistem menampilkan laporan dan dapat tampil ke halaman pdf	System menampilkan laporan dan dapat tampil ke halaman pdf	Valid Alur terlewati
2	1-2-3-4-5-6-7-8-9	Jika data pada bulan atau tahun kosong, system menampilkan flash data dan mengarahkan pengguna kembali ke halaman utama laporan	system menampilkan flash data dan mengarahkan pengguna kembali ke halaman utama laporan	Valid Alur terlewati

## BLACKBOX TESTING

Pada pengujian Black Box digunakan metode Equivalence Partitioning. Equivalence Partitioning merupakan teknik pemilihan data eksperimen yang membagi rentang masukan menjadi kelompok yang sama atau setara. Ide dasarnya adalah jika satu kelas bawaan bekerja dengan baik, kelas serupa lainnya juga akan bekerja dengan baik. Dalam Equivalence Partitioning, inputan dibagi menjadi kelompok atau divisi yang sama dengan karakteristik serupa. Kelompok-kelompok ini mewakili kondisi yang seharusnya memberikan hasil yang sama atau serupa ketika diuji.

### A. Testing Modul Login



Gambar 1 Modul Login

No	Kasus Uji Login	Hasil Yang Diharapkan	Hasil Sesuai Uji Kaus	Keterangan
1	Login dengan user name tanpa password	Gagal Login	Berhasil	Memunculkan pop up password harus diisi
2	Login dengan password tanpa username	Gagal Login	Berhasil	Memunculkan pop up username harus diisi
3	Login dengan username dengan password benar	Berhasil Login	Berhasil	Masuk ke halaman dashboard
4	Login dengan username dan password salah	Gagal Login	Berhasil	Memunculkan pop upusername atau password salah

Pada pengujian modul login dilakukan 4 pengujian yang dimana seluruh pengujian berhasil sesuai hasil yang diharapkan. Sehingga hasil yang dikeluarkan berupa 100% berdasarkan:

$$\text{Persentase Keberhasilan} = \left( \frac{\text{Jumlah Pengujian Berhasil}}{\text{Total Pengujian}} \right) \times 100\%$$

$$\text{Persentase Keberhasilan} = \left( \frac{4}{4} \right) \times 100\%$$

$$\text{Persentase Keberhasilan} = (1) \times 100\% = 100\%$$

## B. Testing Modul Data Akun

Gambar 2 Modul Data Akun

No	Kasus Uji Data Akun	Hasil Yang Diharapkan	Hasil Sesuai Uji Kaus	Keterangan
1	No reff terisi, nama reff dan keterangan kosong	Gagal Input	Berhasil	Pop up nama reff wajib diisi
2	No reff tidak terisi, nama reff terisi dan keterangan kosong	Gagal Input	Berhasil	Pop up no.reff wajib diisi
3	No.reff diisi dengan nilai yang sama	Gagal Input	Berhasil	No reff sudah ada
4	Nama.reff diisi dengan nilai yang sama	Gagal Input	Berhasil	Nama reff sudah ada
5	Nama reff dan no reff tidak diisi	Gagal Input	Berhasil	No reff dan nama reff wajib diisi
6	Nama reff dan no reff diisi dengan nilai yang berbeda	Berhasil Input	Berhasil	Pop up berhasil, data akun berhasil dibuat

Pada pengujian modul Data Aun dilakukan 6 pengujian yang dimana seluruh pengujian berhasil sesuai hasil yang diharapkan. Sehingga hasil yang dikeluarkan berupa 100% berdasarkan :

$$Persentase Keberhasilan = \left( \frac{Jumlah Pengujian Berhasil}{Total Pengujian} \right) \times 100\%$$

$$Persentase Keberhasilan = \left( \frac{6}{6} \right) \times 100\%$$

$$Persentase Keberhasilan = (1) \times 100\% = 100\%$$

### C. Testing Modul Akun

No	Kasus Uji Edit Dan Hapus Data Akun	Hasil Yang Diharapkan	Hasil Sesuai Uji Kaus	Keterangan
1	Edit no reff berbeda, nama reff sama	Gagal Input	Berhasil	Nama reff sudah ada
2	Edit no reff sama , nama reff berbeda	Gagal Input	Berhasil	No reff sudah ada
3	Edit no reff sama dan nama reff sama, keterangan berbeda	Gagal Input	Berhasil	No reff sudah ada dan nama reff sudah ada
4	Edit nama reff dan no reff diubah, keterangan sama	Berhasil Input	Berhasil	Berhasil diubah
5	Edit nama reff dan no reff diubah, keterangan berbeda	Berhasil Input	Berhasil	Berhasil diubah
6	Hapus data dengan tombol hapus	Berhasil Dihapus	Berhasil	Data berhasil dihapus
7	Hapus data dengan tombol cancel	Gagal Dihapus	Berhasil	Data tidak jadi dihapus
8	Hapus data dengan kondisi no reff terdapat pada jurnal	Gagal Dihapus	Berhasil	Pop up tidak bisa dihapus karena data akun terdapat di jurnal umum
9	Hapus data dengan kondisi no reff terdapat pada jurnal dengan kondisi pada jurnal umum sudah dihapus	Gagal Dihapus	Gagal	Pop up tidak bisa dihapus karena data akun terdapat di jurnal umum

Pada pengujian modul Data Akun dilakukan 9 pengujian, dengan hasil 8 pengujian berhasil sesuai harapan dan 1 pengujian gagal, maka :

$$Persentase Keberhasilan = \left( \frac{Jumlah Pengujian Berhasil}{Total Pengujian} \right) \times 100\%$$

$$\text{Persentase Keberhasilan} = \left(\frac{8}{9}\right) \times 100\%$$

$$\text{Persentase Keberhasilan} = (0.888) \times 100\% = 88.8\%$$

Didapatkan hasil berupa 88.8% dalam pengujian modul data akun .

#### D. Testing Modul Jurnal Umum

The screenshot shows the STA application's Jurnal Umum module. The sidebar menu has items: Dashboard, Data Akun, Jurnal Umum (which is selected and highlighted in blue), Buku Besar, Neraca Saldo, and Laporan. The main content area has a purple header 'JURNAL UMUM'. Below it, a sub-header 'Jurnal Umum' with a 'Tambah Jurnal' button. At the top right are date selection dropdowns for 'Januari' and '2018' with a 'Cari' button. A table lists one journal entry: NO. 1, BULAN DAN TAHUN November 2018, with a 'Lihat Jurnal' button. At the bottom left, it says '2023 - Kelompok 5'. The URL 'localhost/sta/index.html' is visible at the bottom left of the browser.

*Gambar 3 Modul Jurnal Umum*

This screenshot shows a detailed view of the Jurnal Umum module. The sidebar menu is identical to the previous screenshot. The main content area has a purple header 'Jurnal Umum'. Below it, a sub-header 'Jurnal Umum'. A table lists multiple journal entries with columns: TANGGAL, NAMA AKUN, REF, DEBET, KREDIT, and ACTION (Edit, Hapus). The entries are: 03 November 2018, Peralatan, 121, Rp. 35.000.000, Rp. 0; 03 November 2018, Modal, 311, Rp. 0, Rp. 35.000.000; 03 November 2018, Kas, 111, Rp. 80.000.000, Rp. 0; 03 November 2018, Modal, 311, Rp. 0, Rp. 80.000.000; 04 November 2018, Beban Sewa, 512, Rp. 6.000.000, Rp. 0; 04 November 2018, Kas, 111, Rp. 0, Rp. 6.000.000; 05 November 2018, Perlengkapan, 113, Rp. 1.900.000, Rp. 0.

*Gambar 4 Modul Jurnal Umum*

No	Kasus Uji Jurnal Umum	Hasil Yang Diharapkan	Hasil Sesuai Uji Kaus	Keterangan
1	Pencarian jurnal umum yang sudah terinput	Gagal Melakukan Pencarian	Berhasil	Memunculkan pop up data jurnal pada

	dengan kondisi bulan salah tahun benar			bulan yang dicari tidak ditemukan
2	Pencarian jurnal umum yang sudah terinput dengan kondisi bulan benar tahun salah	Gagal Melakukan Pencarian	Berhasil	Memunculkan pop up data jurnal pada tahun yang dicari tidak ditemukan
3	Pencarian jurnal umum yang sudah terinput dengan kondisi bulan benar tahun benar	Jurnal Ditemukan	Berhasil	Menampilkan jurnal sesuai kondisi pencarian
4	Nama akun dan saldo tidak diisi	Gagal Input	Berhasil	Nama akun dan saldo wajib diisi
5	Nama akun diisi, saldo tidak diisi	Gagal Input	Berhasil	Saldo wajib diisi
6	Nama akun tidak diisi	Gagal Input	Berhasil	Nama akun wajib diisi
7	Nama akun diisi dan saldo diisi	Berhasil Input	Berhasil	Data jurnak berhasil ditambahkan
8	Edit nama akun, saldo tidak diubah	Berhasil Edit	Berhasil	Data berhasil diubah
9	Edit saldo diubah, nama akun tidak diubah	Berhasil Edit	Berhasil	Data berhasil diubah
10	Edit saldo diubah, akun diubah	Berhasil Edit	Berhasil	Data berhasil diubah
11	Edit saldo dan data akun tidak ubah	Berhasil Edit	Berhasil	Data berhasil diubah
12	Hapus data dengan tombol cancel	Gagal Dihapus	Berhasil	Data tidak jadi dihapus
13	Hapus data dengan tombol hapus	Berhasil Dihapus	Berhasil	Data berhasil dihapus

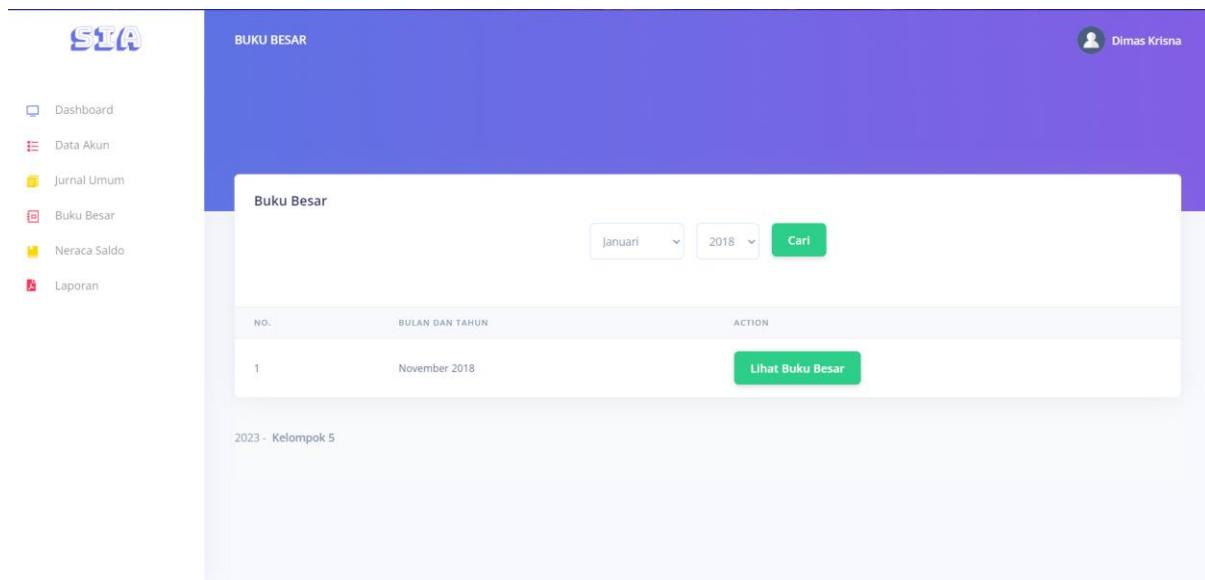
Pada pengujian modul login dilakukan 12 pengujian yang dimana seluruh pengujian berhasil sesuai hasil yang diharapkan. Sehingga hasil yang dikeluarkan berupa 100% berdasarkan:

$$\text{Persentase Keberhasilan} = \left( \frac{\text{Jumlah Pengujian Berhasil}}{\text{Total Pengujian}} \right) \times 100\%$$

$$\text{Persentase Keberhasilan} = \left( \frac{13}{13} \right) \times 100\%$$

$$\text{Persentase Keberhasilan} = (1) \times 100\% = 100\%$$

## E. Testing Modul Buku Besar



Gambar 5 Modul Buku Besar

Gambar 6 Modul Buku Besar

No	Kasus Uji Neraca Saldo	Hasil Yang Diharapkan	Hasil Sesuai Uji Kaus	Keterangan
1	Pencarian buku besar yang sudah terinput dengan kondisi bulan salah tahun benar	Gagal Melakukan Pencarian	Berhasil	Memunculkan pop up data buku besar pada bulan yang dicari tidak ditemukan
2	Pencarian buku besar yang sudah terinput dengan kondisi bulan benar tahun salah	Gagal Melakukan Pencarian	Berhasil	Memunculkan pop up data buku besar pada tahun yang

				dicari tidak ditemukan
3	Pencarian buku besar yang sudah terinput dengan kondisi bulan benar tahun benar	Jurnal Ditemukan	Berhasil	Menampilkan buku besar sesuai kondisi pencarian

Pada pengujian modul login dilakukan 3 pengujian yang dimana seluruh pengujian berhasil sesuai hasil yang diharapkan. Sehingga hasil yang dikeluarkan berupa 100% berdasarkan:

$$\text{Persentase Keberhasilan} = \left( \frac{\text{Jumlah Pengujian Berhasil}}{\text{Total Pengujian}} \right) \times 100\%$$

$$\text{Persentase Keberhasilan} = \left( \frac{3}{3} \right) \times 100\%$$

$$\text{Persentase Keberhasilan} = (1) \times 100\% = 100\%$$

#### F. Testing Modul Neraca Saldo

Gambar 7 Modul Neraca Saldo

NO. AKUN	NAMA AKUN	DEBIT	KREDIT
111	Kas	Rp. 71.400.000	-
112	Piatang	Rp. 200.000	-
113	Perlengkapan	Rp. 750.000	-
121	Peralatan	Rp. 37.000.000	-
122	Akumulasi Peralatan	-	Rp. 250.000
211	Utang Usaha	-	Rp. 1.800.000
311	Modal	-	Rp. 115.000.000
312	Prive	Rp. 750.000	-
411	Pendapatan	-	Rp. 3.450.000
511	Beban Gaji	Rp. 900.000	-
512	Beban Sewa	Rp. 6.500.000	-
513	Beban Penyusutan Peralatan	Rp. 250.000	-

Gambar 8 Modul Neraca Saldo

No	Kasus Uji Neraca Saldo	Hasil Yang Diharapkan	Hasil Sesuai Uji Kaus	Keterangan
1	Pencarian neraca saldo yang sudah terinput dengan kondisi bulan salah tahun benar	Gagal melakukan pencarian	Berhasil	Memunculkan pop up data neraca saldo pada bulan yang dicari tidak ditemukan
2	Pencarian neraca saldo yang sudah terinput dengan kondisi bulan benar tahun salah	Gagal melakukan pencarian	Berhasil	Memunculkan pop up data neraca saldo pada tahun yang dicari tidak ditemukan
3	Pencarian neraca saldo yang sudah terinput dengan kondisi bulan benar tahun benar	Jurnal ditemukan	Berhasil	Menampilkan neraca saldo sesuai kondisi pencarian

sesuai hasil yang diharapkan. Sehingga hasil yang dikeluarkan berupa 100% berdasarkan:

$$Persentase Keberhasilan = \left( \frac{\text{Jumlah Pengujian Berhasil}}{\text{Total Pengujian}} \right) \times 100\%$$

$$Persentase Keberhasilan = \left( \frac{3}{3} \right) \times 100\%$$

$$Persentase Keberhasilan = (1) \times 100\% = 100\%$$

## G. Testing Modul Laporan

Gambar 9 Modul Laporan

No	Kasus Uji Laporan	Hasil Yang Diharapkan	Hasil Sesuai Uji Kaus	Keterangan
1	Pencarian laporan yang sudah terinput dengan kondisi bulan salah tahun benar	Gagal melakukan pencarian	Berhasil	Memunculkan pop up data laporan pada bulan yang dicari tidak ditemukan
2	Pencarian laporan yang sudah terinput dengan kondisi bulan benar tahun salah	Gagal melakukan pencarian	Berhasil	Memunculkan pop up data laporan pada tahun yang dicari tidak ditemukan
3	Pencarian laporan yang sudah terinput dengan kondisi bulan benar tahun benar	Jurnal ditemukan	Berhasil	Menampilkan laporan sesuai kondisi pencarian
4	Pencetakan laporan	Laporan terdonwload dan siap untuk dicetak	Berhasil	Menampilkan halaman laporan yang sap dicetak

sesuai hasil yang diharapkan. Sehingga hasil yang dikeluarkan berupa 100% berdasarkan:

$$\text{Persentase Keberhasilan} = \left( \frac{\text{Jumlah Pengujian Berhasil}}{\text{Total Pengujian}} \right) \times 100\%$$

$$\text{Persentase Keberhasilan} = \left( \frac{4}{4} \right) \times 100\%$$

$$\text{Persentase Keberhasilan} = (1) \times 100\% = 100\%$$