

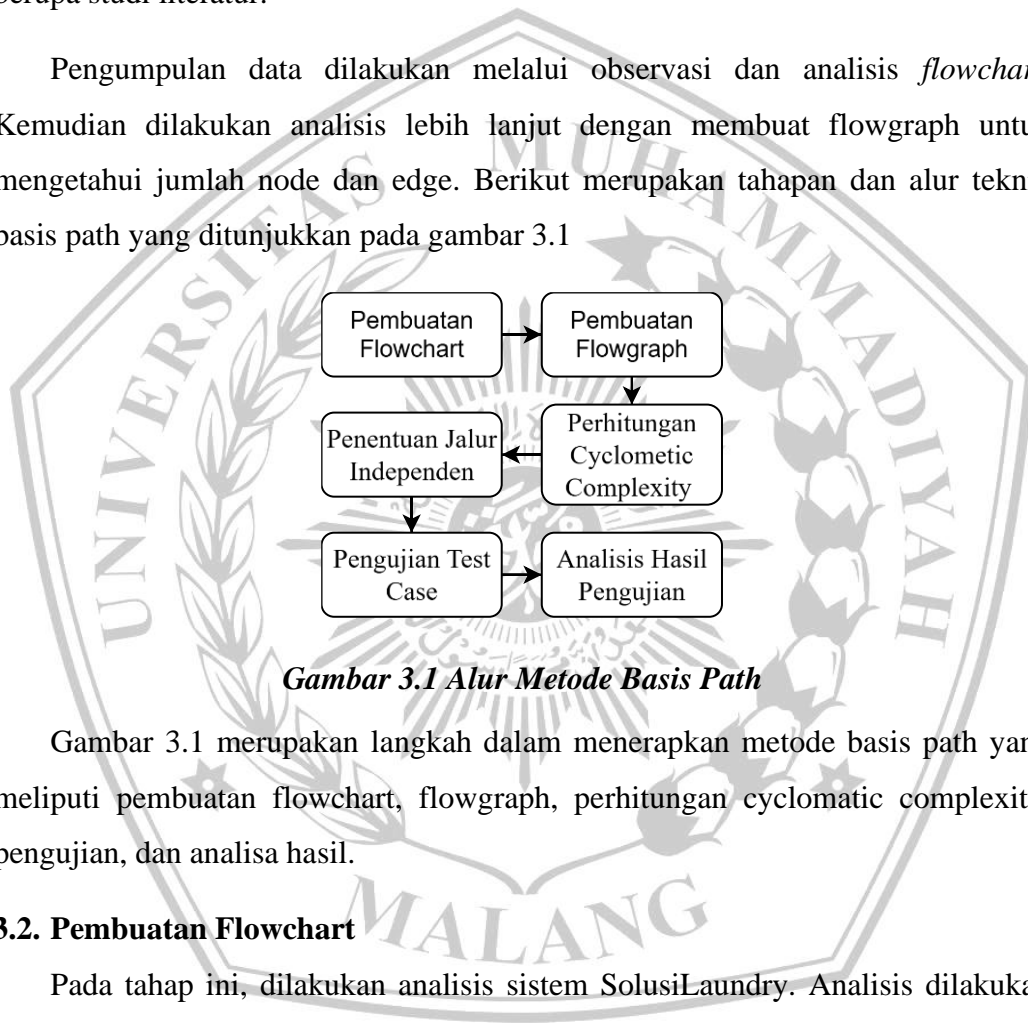
## BAB III

### METHODOLOGI

#### 3.1. Metodologi Penelitian

Penelitian ini memanfaatkan desain pendekatan Teknik basis path dalam menganalisis sistem dan pengujian dengan menggabungkan pendekatan kualitatif berupa studi literatur.

Pengumpulan data dilakukan melalui observasi dan analisis *flowchart*. Kemudian dilakukan analisis lebih lanjut dengan membuat *flowgraph* untuk mengetahui jumlah node dan edge. Berikut merupakan tahapan dan alur teknik basis path yang ditunjukkan pada gambar 3.1



**Gambar 3.1 Alur Metode Basis Path**

Gambar 3.1 merupakan langkah dalam menerapkan metode basis path yang meliputi pembuatan flowchart, flowgraph, perhitungan cyclomatic complexity, pengujian, dan analisa hasil.

#### 3.2. Pembuatan Flowchart

Pada tahap ini, dilakukan analisis sistem SolusiLaundry. Analisis dilakukan dengan menggambarkan *flowchart* yang memrepresentasikan alur kerja sistem yang akan diuji. *Flowchart* adalah alur logika yang digambarkan dengan simbol standar dan panah untuk mempresentasikan sistem [19]. *Flowchart* menggunakan bentuk persegi panjang untuk menggambarkan proses, oval untuk menggambarkan start atau end, dan garis panah untuk alur arah.

### 3.3. Pembuatan Flowgraph

Pada tahap ini, dilakukan analisis lanjutan sistem SolusiLaundry. Analisis dilakukan dengan mengubah flowchart yang memrepresentasikan alur kerja sistem yang akan diuji menjadi flowgraph. Notasi pada flowgraph digambarkan dengan lingkaran (node) mewakili proses program dan alur panah (edge) sebagai alur penghubung antar instruksi.

### 3.4. Perhitungan Cyclometix Complexity

Pada tahap ini, dilakukan perhitungan kompleksitas sistem dengan menggunakan teknik Cyclometix Complexity. Hasil dari perhitungan ini dapat menjadi jumlah minimal kasus uji yang diperlukan dan menjadi indikator kompleksitas logika sistem [20]. Perhitungan nilai kompleksitas menggunakan teknik cyclomatic complexity seperti berikut:

$$V(G) = E - N + 2 \quad (1)$$

Dimana:

$V(G)$  = Jumlah kompleksitas

$E$  = Jumlah Edge pada flowgraph

$N$  = Jumlah Node pada flowgraph

Persamaan (1) adalah formula untuk mendapatkan nilai kompleksitas logika sistem. Lingkaran (node) mewakili proses intruksi program dan garis panah penghubung (edge) untuk menggambarkan alur antar instruksi yang dieksekusi.

### 3.5. Penentuan Jalur Independen

Pada tahap ini, dilakukan analisis jalur independent yang hendak diuji berdasarkan nilai dari kompleksitas sistem. Jalur independent merupakan alur program yang dimulai dari node awal hingga node akhir dengan ketentuan melalui edge baru dengan jalur yang belum pernah dilalui sebelumnya. Penentuan jalur independent dilakukan dengan memperhatikan aspek-aspek khusus dari struktur dan fitur sistem.

### 3.6. Pengujian Test Case

Test case dibuat untuk menguji atau mengeksekusi semua jalur independen (alur logika) yang sudah dibuat. Kasus uji yang dilakukan sekurang-kurangnya berjumlah sesuai dengan nilai dari perhitungan *cyclomatic complexity*. Pengujian dilakukan dengan melakukan testing secara manual terhadap skenario yang telah dibuat pada website [solusilaundry.vercel.app](https://solusilaundry.vercel.app).

### 3.7. Analisa Hasil

Tahap terakhir, penulis menganalisis hasil pengujian test case. Hasil analisis ini yang kemudian menjadi bahan pertimbangan apakah aplikasi sudah layak dan sesuai dengan fungsionalitasnya.

