

BAB 3

METODE PENELITIAN

Penelitian ini bertujuan untuk menguji fungsionalitas sistem JB Class menggunakan metode *Black Box Testing* dengan teknik *Boundary Value Analysis* dan *Equivalence Partitioning*. Metode ini digunakan untuk mengevaluasi inputan dan *output* dari sistem tanpa melihat kode sumber, sehingga dapat diidentifikasi kesalahan yang mungkin terjadi dalam berbagai skenario penggunaan. Berikut ini dijelaskan mengenai tempat pelaksanaan, bahan dan alat yang digunakan, serta tahapan penelitian dalam pengujian sistem JB Class.

3.1 WAKTU DAN TEMPAT

Penelitian ini dilaksanakan mulai bulan Februari 2025 hingga Juni 2025 di Balai Tekkomdik DIY yang berlokasi di Jl. Kenari No.2, Semaki, Kecamatan Umbulharjo, Kota Yogyakarta.

3.2 ALAT DAN BAHAN

3.2.1 Alat Penelitian

Dalam penelitian ini, terdapat beberapa alat yang digunakan, yang terdiri dari perangkat keras dan perangkat lunak sebagai berikut:

a. Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini adalah laptop Asus X415 dengan prosesor Intel(R) Core(TM) i3-1115G4 dan ram 12,0 GB, serta mouse Logitech M170 *wireless*.

b. Perangkat Lunak

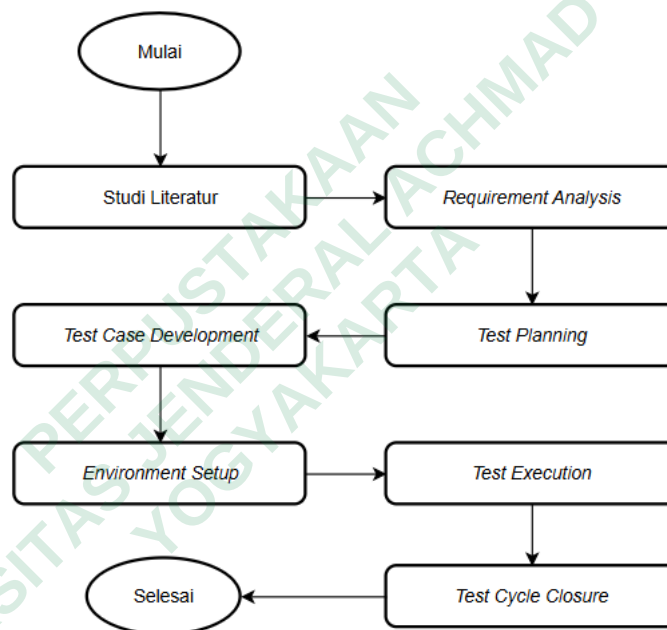
Perangkat lunak yang digunakan dalam penelitian ini mencakup sistem operasi Windows 10, Microsoft Word 2019, Microsoft Excel 2019, Google *Spreadsheets*, *Mendeley Reference Manager*, browser Google *Chrome* dan Microsoft *Edge*.

3.2.2 Bahan Penelitian

Bahan yang dan sumber daya yang digunakan dalam penelitian ini meliputi literatur dari jurnal, Google *Books*, dan situs web, aplikasi berbasis web JB Class, serta panduan pengguna JB Class.

3.3 TAHAPAN PENELITIAN

Tahapan penelitian dalam proses pengujian sistem JB Class dilakukan dengan mengikuti alur STLC. Berikut merupakan gambaran dari alur STLC yang disajikan pada Gambar 5.1 (Nordeen, 2020):



Gambar 3. 1. Alur STLC

3.3.1 Studi Literatur

Tahap awal penelitian ini dilakukan dengan mengumpulkan data dan informasi dari berbagai sumber, seperti buku elektronik, jurnal, Google *Books*, dan literatur terkait. Fokus utama dalam studi ini mencakup pengujian *Black Box*, metode *Equivalence Partitioning*, *Boundary Value Analysis*, serta implementasi pengujian pada sistem pembelajaran daring. Selain itu, studi juga mencakup penelitian terdahulu dan studi kasus yang relevan untuk memperkuat dasar teori dalam penelitian ini.

3.3.2 *Requirement Analysis*

Pada tahap ini, dilakukan analisis kebutuhan sistem JB Class dengan mengumpulkan spesifikasi teknis dan dokumen terkait. Proses ini mencakup pemahaman terhadap fitur dan fungsionalitas sistem yang akan diuji. Selain itu, diskusi dengan pemangku kepentingan atau tim pengembang dilakukan untuk memperoleh wawasan lebih mendalam mengenai sistem JB Class, sehingga pengujian dapat disesuaikan dengan kebutuhan aktual pengguna.

3.3.3 *Test Planning*

Setelah analisis kebutuhan selesai, tahap selanjutnya adalah perencanaan pengujian berdasarkan hasil analisis, studi literatur, dan dokumen teknis aplikasi. Pada tahap ini, disusun strategi pengujian yang meliputi penentuan cakupan pengujian, metode yang digunakan, serta jadwal pengujian. Perencanaan ini bertujuan untuk memastikan pengujian berjalan secara sistematis dan sesuai dengan tujuan penelitian.

Metode pengujian yang diterapkan dalam penelitian ini adalah *Black Box Testing* dengan teknik *Boundary Value Analysis* dan *Equivalence Partitioning*. Pengujian dilakukan pada fitur-fitur JB Class yang melibatkan proses *input*, guna memastikan bahwa sistem berfungsi sesuai dengan spesifikasi yang telah ditetapkan.

Pada teknik *Boundary Value Analysis*, setiap inputan akan diuji dengan enam *test case*, yaitu nilai tepat di bawah batas minimum (Min-1), nilai pada batas minimum (Min), nilai tepat di atas batas minimum (Min+1), nilai tepat di bawah batas maksimum (Max-1), nilai pada batas maksimum (Max), dan nilai tepat di atas batas maksimum (Max+1). Sedangkan dalam teknik *Equivalence Partitioning*, setiap *input* akan diuji menggunakan partisi valid (dalam rentang yang diizinkan) dan tidak valid (di luar rentang yang diizinkan (Desai & Srivastava, 2016)).

3.3.4 Test Case Development

Pada tahap ini, dilakukan penyusunan *test case* berdasarkan kebutuhan sistem yang telah dikumpulkan sebelumnya serta metode pengujian yang telah ditetapkan. Pembuatan *test case* akan mempertimbangkan skenario penggunaan yang relevan untuk memastikan cakupan pengujian yang optimal. Selain itu, tim pengembang akan dilibatkan dalam proses ini guna memastikan bahwa *test case* yang dibuat sesuai dengan spesifikasi sistem JB Class.

3.3.5 Environment Setup

Tahap ini bertujuan untuk menyiapkan dan mengonfigurasi lingkungan pengujian agar sesuai dengan kebutuhan sistem. Persiapan mencakup pengaturan perangkat keras, perangkat lunak, jaringan, data pengujian, serta sistem pendukung lainnya yang diperlukan untuk menjalankan *test case* secara efektif. Pengujian dilakukan pada platform JB Class yang berbasis web dalam kondisi terhubung dengan koneksi internet untuk menyesuaikan karakteristik sistem yang berjalan secara daring.

3.3.6 Test Execution

Tahap utama dalam penelitian ini adalah melakukan pengujian terhadap sistem JB Class. Tujuan dari tahap ini adalah untuk mengevaluasi apakah fitur-fitur yang terdapat pada sistem telah berjalan sesuai dengan fungsionalitas yang diharapkan, serta untuk mengidentifikasi adanya kesalahan atau ketidaksesuaian pada sistem. Pengujian dilakukan dengan mengacu pada *test case* yang telah dirancang secara sistematis pada tahap sebelumnya.

Pelaksanaan pengujian dilakukan dengan mengikuti langkah-langkah yang telah ditetapkan dalam *test case*, dimulai dari kondisi awal (*pre-condition*), data uji, langkah pengujian (*test steps*), hingga pemeriksaan hasil keluaran yang diharapkan (*expected result*). Selama proses pengujian, setiap hasil aktual (*actual result*) yang diperoleh dicatat dan dibandingkan dengan hasil yang diharapkan.

3.3.7 Test Cycle Closure

Tahap akhir penelitian ini melibatkan analisis dan evaluasi hasil pengujian. Setiap *test case* yang telah dijalankan akan dianalisis untuk mengetahui kinerja sistem JB Class telah berfungsi dengan baik atau masih terdapat kekurangan yang perlu diperbaiki. Hasil pengujian akan disajikan dalam bentuk persentase validitas dan *invaliditas test case*. Berdasarkan hasil pengujian tersebut, akan disusun rekomendasi perbaikan yang disajikan dalam bentuk tulisan naratif, sesuai dengan temuan permasalahan pada fitur-fitur yang mengalami kegagalan atau ketidaksesuaian. Persentase validitas dan *invaliditas test case* dihitung menggunakan rumus berikut (Suharyono et al., 2024):

$$Test\ Case\ Valid = \left(\frac{a}{b}\right) \times 100\% \quad (1)$$

$$Test\ Case\ Invalid = \left(\frac{c}{b}\right) \times 100\% \quad (2)$$

Keterangan:

a = Jumlah *test case* yang valid

b = Jumlah *test case* yang dilakukan

c = Jumlah *test case* yang tidak valid