|  |  |
| --- | --- |
| **PENERAPAN JEST DAN SUPERTEST PADA LIBRARY NPM UNTUK MENINGKATKAN OTOMATISASI PENILAIAN SOURCE CODE JAVASCRIPT** | |
| **PROPOSAL SKRIPSI** | |
|  | |
| **Rahmat Sunjani**  **55201120030** | |
| **PROGRAM STUDI TEKNIKI NFORMATIKA**  **FAKULTAS ILMU KOMPUTER DAN INFORMATIKA**  **UNIVERSITAS NURTANIO BANDUNG**  **2023** | |
| Kepada :  Yth. Bpk. Suharjanto Utomo, S.Si., M.T.  Ketua Program Studi Teknik Informatika (S1)  di tempat | |
| Dengan Hormat,  Dengan ini saya yang bertanda tangan di bawah ini bermaksud mengajukan proposal Skripsi :  Nama : Rahmat Sunjani  NPM : 55201120030  Kelas : IF’20 A  Telephone : 083174506600  IPK : -  SKS : - | |
| Saya menyatakan bersedia dan sanggup menyelesaikan skripsi saya sesuai dengan peraturan yang telah ditetapkan oleh pihak Universitas Nurtanio apabila proposal skripsi ini disetujui. Sebagai bahan pertimbangan, dengan ini saya lampirkan proposal skripsi saya yang berjudul “PENERAPAN JEST DAN SUPERTEST PADA LIBRARY NPM UNTUK MENINGKATKAN OTOMATISASI PENILAIAN SOURCE CODE JAVASCRIPT” Demikianlah surat pengajuan proposal skripsi ini saya buat, atas perhatian dan kebijaksanaan Bapak/Ibu, saya ucapkan terima kasih. | |
|  | Bandung, 15 Desember 2023  Hormat saya |
|  |  |
|  | **Rahmat Sunjani**  **55201120030** |
| **LEMBAR PERSETUJUAN**  **PROPOSAL SKRIPSI** | |
| Nama : Rahmat Sunjani  NPM : 55201120030  Kelas : IF’20 A  Telephone : 083174506600  Judul Tema : PENERAPAN JEST DAN SUPERTEST PADA LIBRARY NPM UNTUK MENINGKATKAN OTOMATISASI PENILAIAN SOURCE CODE JAVASCRIPT  Tanggal Persetujuan : | |
| Menyetujui,  Ketua Program Studi Teknik Informatika, | |
|  | |
| **(Suharjanto Utomo, S.Si., MT)** | |

**ABSTRAK**

Tujuan.metode(cara meneliti, desain, subject data, yang digunakan, metode pengumpulan data,analisis data).Hasil.kesimpulan.keyword

# PENDAHULUAN

## Latar Belakang

Kode Sumber adalah komponen dasar dari program komputer yang dibuat oleh seorang programmer, yang sering kali ditulis dalam bentuk fungsi, deskripsi, definisi, pemanggilan, metode, dan pernyataan operasional lainnya. Kode ini dirancang agar dapat dibaca manusia dan diformat dengan cara yang dapat dimengerti oleh pengembang dan pengguna lain[1]. Namun tidak jarang bahwa sebuah *source code* sering memiliki error atau struktur yang berantakan.

Oleh karena itu, Skripsi ini bertujuan untuk menjawab kebutuhan yang akan menambah pemahaman tentang efisiensi dan kualitas sebuah *source code*. Melalui penelitian ini, diharapkan dapat menemukan tools yang dapat melakukan automasi penilaian kode sumber sehingga dapat meningkatkan dalam pembelajaran tentang pemrograman.

Dengan menyusun latar belakang ini, penelitian ini menciptakan alat yang berguna terutama bagi mereka yang menyediakan materi pembelajaran, seperti Dosen, Guru, Sumber Kode Pendidikan online, dan yang terkait dengan pendidikan mengenai pemrograman. Alat ini dirancang untuk efisiensi dan meminimalkan waktu yang dibutuhkan untuk mengevaluasi jawaban tes secara cepat tanpa harus melakukannya satu per satu sehingga kedepannya tools ini akan sangat membantu dalam pengembangan tentang otomatisasi cek *source code*.

## Rumusan masalah

Adapun rumusan masalah dari penelitian ini, antara lain :

* 1. Mengapa perancangan penilaian otomatis kode dianggap penting?
  2. Bagaimana cara praktis Jest dan Supertest pada library NPM diterapkan untuk mengevaluasi kualitas kode Node.js?
  3. Apa manfaat yang dapat diharapkan dari implementasi otomasi penilaian kode ini dalam pengembangan perangkat lunak?

## Batasan masalah

Pada penelitian ini adapun batasan masalahnya, antara lain :

1. Pengujian ini akan dilakukan dengan bahasa pemrograman Javascript.
2. Pengujian berfokus pada mengecek *source code* Javascript.
3. Pembuatan library akan didukung dengan Method Black Box dan White Box serta dengan dukungan library Jest dan Supertest.

## Tujuan penelitian

Adapun tujuan dari penelitian ini, antara lain :

1. Mengetahui nilai *source code* yang sesuai apa yang diberikan penguji.
2. Memastikan tidak terjadinya duplikat *source code*.
3. Memastikan untuk memudahkan operasi penggunaan library bagi penguji.

## Rencana dan jadwal kegiatan

Tabel 1. Rencana dan Jadwal Kegiatan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Kegiatan | Bulan | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Literatur Review |  |  |  |  |  |  |
| Analisis & Perancangan Sistem |  |  |  |  |  |  |
| Pengengembangan Pembuatan Library |  |  |  |  |  |  |
| Testing Perancangan Library |  |  |  |  |  |  |
| Implementasi Library |  |  |  |  |  |  |
| Evaluasi & Pembarahuan System |  |  |  |  |  |  |

# TEORI DASAR

## Source Code

Source Code atau kode sumber adalah dasar dari rancangan suatu program yang berisi kumpulan baris teks instruksi dan kode-kode fungsi yang mengkomunikasikan suatu perintah yang harus dijalankan oleh program agar program tersebut berfungsi sesuai tujuan perancangan. Di dalam bidang pemrograman, source code merupakan himpunan satu kesatuan berkas atau file yang berisi kode-kode perintah dalam suatu program[2].

## NPM

NPM (Node Package Manager) adalah manajer paket yang banyak digunakan untuk JavaScript yang menyederhanakan proses pengelolaan ketergantungan dalam proyek pengembangan web. NPM memungkinkan pengembang untuk menginstal, memperbarui, dan mengelola pustaka dan alat pihak ketiga dengan mudah. NPM menyediakan ekosistem paket yang luas, memungkinkan pengembang untuk memanfaatkan solusi yang ada dan mempercepat proses pengembangan mereka. Dokumentasi yang disediakan oleh NPM menawarkan panduan komprehensif tentang instalasi, konfigurasi, dan penggunaan paket, menjadikannya sumber daya yang berharga bagi para pengembang[3].

## Node.js

Node.js adalah lingkungan runtime JavaScript sisi server yang memungkinkan pengembang membangun aplikasi web yang dapat diskalakan dan berkinerja tinggi. Node.js menggunakan model I/O yang digerakkan oleh peristiwa dan tidak memblokir, sehingga sangat cocok untuk menangani permintaan yang bersamaan. Node.js memiliki ekosistem modul yang luas yang tersedia melalui NPM, yang memungkinkan pengembang untuk memperluas fungsionalitasnya. Dokumentasi resmi Node.js mencakup berbagai aspek Node.js, termasuk API, modul, dan praktik terbaiknya. Dokumentasi ini berfungsi sebagai referensi komprehensif bagi para pengembang yang bekerja dengan Node.js[3].

## Jest

Jest adalah framework pengujian JavaScript yang dikembangkan oleh Facebook. Ini bekerja di luar kotak dengan konfigurasi minimal dan memiliki in-built test runner, perpustakaan penegasan dan dukungan mocking[4].

## Supertest

Library untuk menguji server HTTP Node.js. Ini memungkinkan kami untuk mengirim permintaan HTTP secara terprogram seperti GET, POST, PATCH, PUT, DELETE ke server HTTP dan mendapatkan hasil[4].

## Black Box

Pengujian Black Box adalah metode di mana fungsionalitas aplikasi perangkat lunak dievaluasi tanpa melihat kode internalnya. Metode ini memeriksa apakah perangkat lunak berperilaku seperti yang diharapkan berdasarkan persyaratan, masukan, dan keluaran. Begini cara kerjanya bekerja: penguji memberikan input ke perangkat lunak dan mengamati output, membandingkannya dengan hasil yang diharapkan. Ini membantu memastikan bahwa perangkat lunak memenuhi fungsionalitas yang diinginkan tanpa perlu mengetahui bagaimana perangkat lunak tersebut dibuat secara internal[5].

## White Box

Metode Pengujian “White Box” adalah metode pengujian yang berfokus pada input dan output dari perngkat lunak yang dibuat oleh seorang programmer. Dengan cara menganalisa dan meneliti langkah-langkah struktur internal dan kode dari sebuah sistem, implementasi, alur data dan kemungkinan kesalahan dari sebuah sistem yang sedang dibangun[6].

# PERANCANGAN SISTEM DAN ALUR PEMODELAN

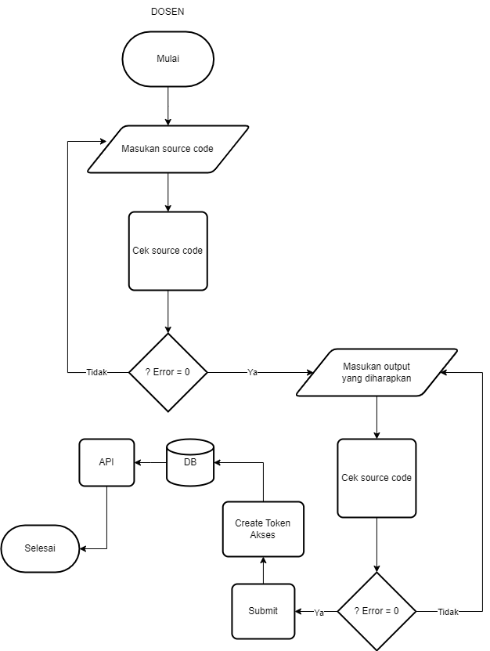
Dalam perancangan sistem, langkah awal melibatkan pembuatan diagram alur proses sistem dan rencana kebutuhan data, termasuk ilustrasi proses pengolahan data. Skenario pengujian disusun untuk memastikan kinerja sistem. Pemilihan metode perancangan sistem, seperti Model Waterfall, diintegrasikan dengan ilustrasi proses. Pada tahap pemodelan, diagram alur pembuatan model dengan rencana tahapan dan skenario pengujian diterapkan. Metode penelitian juga dipilih dan diintegrasikan dalam alur pemodelan. Ini memberikan gambaran yang jelas dan singkat dalam konteks penelitian atau skripsi yang dilakukan.

## Perancangan Sistem

### Flowchart Sistem

Flowchart sistem adalah flowchart yang menampilkan tahapan atau proses kerja yang sedang berlangsung di dalam sistem secara menyeluruh. Selain itu flowchart sistem juga menguraikan urutan dari setiap prosedur yang ada di dalam sistem[7].

Gambar 3. Flowchart Sistem Dosen



Penjelasan Flowchart Sistem Dosen :

**Penguji Memasukan Source Code**

Dosen akan menyertakan source code yang akan menjadi pedoman penilaian bagi proyek ini.

**Cek Source Code**

Pendekatan yang kami terapkan sebelum mengirimkan source code adalah dengan melakukan pengecekan awal menggunakan library ESLint. Langkah ini kami ambil untuk memastikan kualitas kode dan mengurangi potensi kesalahan. Jika dalam proses pengecekan ditemukan kesalahan, kami akan memberikan kesempatan untuk melakukan koreksi sebelum melanjutkan ke tahap berikutnya. Namun, jika pengecekan berjalan lancar dan kode telah memenuhi standar kualitas, kami akan melanjutkan ke tahap selanjutnya, yaitu memasukkan inputan yang diharapkan oleh penguji.

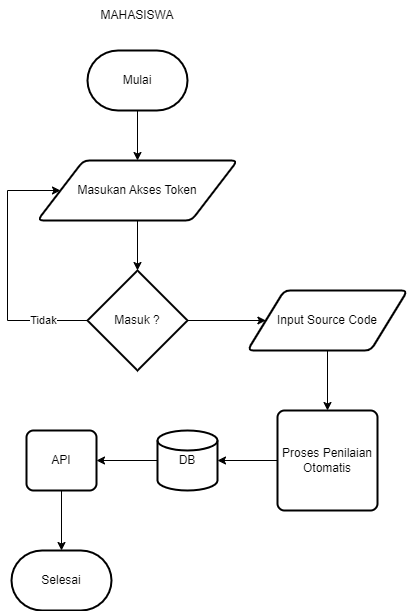
**Masukan Output Yang Diharapkan**

Penguji akan melanjutkan evaluasi setelah sebelumnya hanya memeriksa sejauh mana kesamaan persentase antara source code dengan kode penguji. Dalam penilaian kedua, mereka akan menilai hasil output dengan mempertimbangkan sejauh mana kesesuaian dengan yang diinginkan. Proses ini bertujuan untuk memastikan bahwa setiap aspek telah diperhatikan secara cermat. Jika terdapat ketidaksesuaian, akan diberikan umpan balik konstruktif dan kesempatan untuk penyempurnaan melalui revisi input. Namun, jika berhasil memenuhi kriteria, proyek tersebut akan diteruskan ke menu "submit" sebagai tanda persetujuan evaluasi.

**Submit**

Setelah penyelesaian, data akan disubmit dan disimpan dalam database. Sebelum dimasukkan ke database, akan otomatis menghasilkan akses token yang akan digunakan oleh mahasiswa untuk mendapatkan soal. Setelah tersimpan dalam database, semua informasi akan terhubung melalui API untuk mempermudah akses.

Gambar 3. Flowchart Sistem Mahasiswa



Penjelasan Flowchart Sistem Mahasiswa :

**Masukan Akses Token**

Penguji harus memiliki akses token yang telah disiapkan sebelumnya untuk memasukkan jawaban. Jika penggunaan token tidak berhasil, akan diminta untuk dimasukkan kembali; namun jika berhasil, pengguna akan diarahkan langsung ke menu input jawaban source code yang harus diisi oleh mahasiswa.

**Input Source Code**

Dalam skripsi ini, input source code tidak dapat direvisi, melainkan mahasiswa akan menerima penilaian secara langsung. Setelah pengumpulan, source code akan menjalani evaluasi otomatis, dan hasilnya tidak akan diakses oleh mahasiswa. Setelah proses evaluasi selesai, data akan disimpan langsung ke dalam database dan dikembalikan melalui integrasi API.

**Proses Penilaian Otomatis**

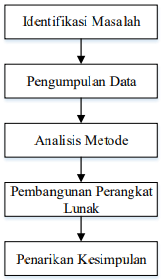
Pada proses evaluasi source code otomatis, akan dilibatkan beberapa metode yang akan diaplikasikan beserta penerapan beberapa perpustakaan (library) dari Node.js yang akan dijelaskan dalam Alur Pemodelan.

## Alur Pemodelan

### Alur Penelitian

Alur penelitian pada penelitian ini mencakup lima tahap, yaitu identifikasi masalah, pengumpulan data, analisis metode, pembangunan perangkat lunak, dan penarikan kesimpulan. Alur penelitian dapat dilihat pada Gambar 3.3.

Gambar 3. Alur Penelitian



Penjelasan dari alur penelitian pada Gambar 3.3 adalah sebagai berikut.

**Identifikasi Masalah**

Tahap identifikasi masalah merupakan proses pengamatan terhadap penelitian yang sudah pernah dilakukan sebelumnya untuk menentukan kebutuhan dan tujuan sistem yang akan dicapai.

**Pengumpulan Data**

Pada penelitian ini, pengumpulan data yang dilakukan adalah studi literator baik secara tercetak maupun elektronik.

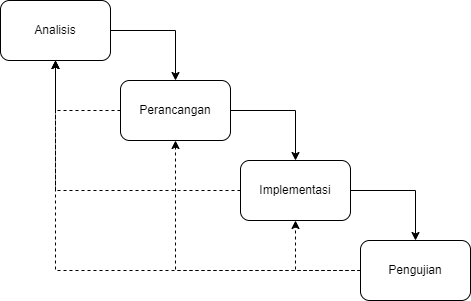
**Analisis Metode**

Pada tahap analisis metode, metode yang digunakan dalam penelitian ini akan dianalisis, mulai dari tahap preprocessing, proses analisis, dan proses translasi.

**Pembangunan Perangkat Lunak**

Pada penelitian ini, pembangunan perangkat lunak yang digunakan adalah model waterfall. Model waterfall merupakan metode pembangunan perangkat lunak yang bersifat sekuensial dalam tiap prosesnya[8]. Alur dari model waterfall dapat dilihat pada Gambar 3.4.

Gambar 3. Model Waterfall



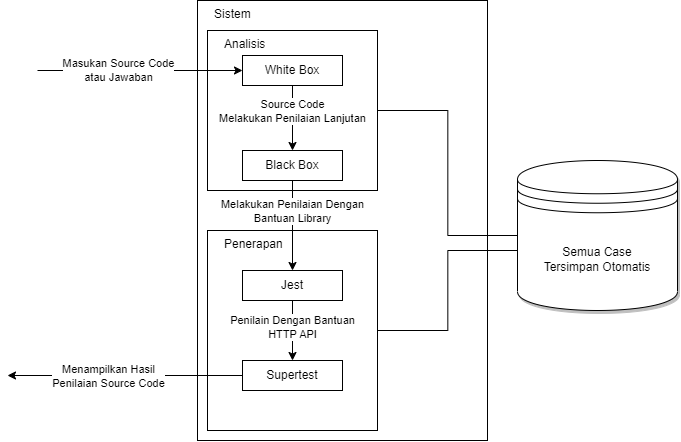
Penarikan Kesimpulan

Pada tahap penarikan kesimpulan merupakan penjelasan hasil dari penelitian yang sudah dilakukan.

### Gambaran Umum Sistem

Dalam penelitian ini, sistem yang dikembangkan memiliki kemampuan untuk mengevaluasi source code yang ditulis dalam bahasa Javascript. Sistem ini terdiri dari dua tahap utama, yaitu analisis penerapan metode dan penerapan metode atau library Node.js. Proses analisis melibatkan langkah-langkah implementasi Black Box dan White Box, sementara proses penerapan metode mencakup tahap pembuatan kode yang didukung dengan library Jest dan Supertest.

Gambar 3. Blok Diagram Sistem Keseluruhan



**DAFTAR PUSTAKA**

[1] Scott Wallask, “Source Code,” www.techtarget.com. Diakses: 13 Desember 2023. [Daring]. Tersedia pada: https://www.techtarget.com/searchapparchitecture/definition/source-code

[2] Dwi Arizki Verdianto, “Source Code,” teknogram.id. Diakses: 14 Desember 2023. [Daring]. Tersedia pada: https://teknogram.id/kamus/source-code/

[3] N. A. Alhazmy, Z. N. Chandra, P. Atmadiputra, dan Y. Triyana, “Building a Comprehensive Content Management System with NPM, Vue.js, Node.js, Postgresql, and Strap,” 2023.

[4] Chinedu Orie, “Testing NodeJs/Express API with Jest and Supertest,” dev.to. Diakses: 15 Desember 2023. [Daring]. Tersedia pada: https://dev.to/nedsoft/testing-nodejs-express-api-with-jest-and-supertest-1km6

[5] Prof. S.K.Totade, Trupti Tayde, dan Pranali Dhole, “Black Box Testing,” *f Innovations in Engineering and Technology (IRJIET)*, vol. Volume 7, 2023, doi: https://doi.org/10.47001/IRJIET/2023.710089.

[6] A. Fahma Rosyada, I. Sukirman, M. Afrizal Nur, dan A. Saifudin, “BULLET : Jurnal Multidisiplin Ilmu Pengujian Sistem Informasi Aplikasi Perpustakaan basis Website Menggunakan White Box Testing,” *BULLET : Jurnal Multidisiplin Ilmu*, vol. Volume 1, 2022, [Daring]. Tersedia pada: https://journal.mediapublikasi.id/index.php/oktal

[7] Rony Setiawan, “Flowchart Adalah: Fungsi, Jenis, Simbol, dan Contohnya,” www.dicoding.com. Diakses: 15 Desember 2023. [Daring]. Tersedia pada: https://www.dicoding.com/blog/flowchart-adalah/

[8] S. P. Roger dan R. M. Bruce, *Software engineering: a practitioner’s approach*. McGraw-Hill Education, 2015. Diakses: 15 Desember 2023. [Daring]. Tersedia pada: https://dspace.agu.edu.vn/handle/agu\_library/13103

LAMPIRAN