

PENERAPAN METODE PROTOTYPE DALAM PENGEMBANGAN WEBSITE KONVERSI GAMBAR VEKTOR KE MODEL 3D

Nia Yuningsih¹⁾, Zaidan Ramadhan Rachman²⁾, Lely Prananingrum³⁾

1. Fakultas Ilmu Komputer dan Teknologi Industri, Universitas Gunadarma
email: niaayuningsih802@gmail.com
2. Fakultas Ilmu Komputer dan Teknologi Industri, Universitas Gunadarma
email: zaidanramadhanr9@gmail.com
3. Fakultas Ilmu Komputer dan Teknologi Industri, Universitas Gunadarma
email: lelyprana@gmail.com

Abstract

This research aims to design and build a static website capable of automatically converting SVG vector images into 3D objects. The main problem addressed is the lack of user-friendly and efficient web-based tools for converting SVG files into 3D models, which are important for 3D designers and 3D printing. The development applies the Prototype method, allowing initial prototypes to be tested and refined based on user feedback. The website is developed using JavaScript with the Three.js library for 3D object processing and visualization, and Vite as a modern development tool to ensure responsiveness. Users are able to upload SVG files, adjust object thickness, preview the conversion in real-time, and export the 3D model in STL format. The results show that this website provides a practical and interactive solution for users who require fast conversion from SVG images to 3D objects through a web platform, accessible at <https://sv3dkonversi.xyz>. User Acceptance Testing was conducted and achieved a score of 82.2%, indicating that users can operate the website effectively in terms of layout, interface, system, and ease of use. This demonstrates the potential of the website to support 3D designers in digital manufacturing and 3D printing workflows.

Kata Kunci : Conversion, 3D, Javascript, Multimedia, Website

A. PENDAHULUAN

Perubahan signifikan telah terjadi akibat kemajuan teknologi *digital* di sejumlah disiplin ilmu, termasuk desain grafis dan manufaktur. Gambar vektor, khususnya dalam format *SVG (Scalable Vector Graphics)*, semakin banyak digunakan karena kemampuannya untuk mempertahankan kualitas gambar pada berbagai ukuran. *SVG* banyak dimanfaatkan dalam pembuatan logo, ikon, ilustrasi, hingga desain produk yang

membutuhkan presisi tinggi [1]. Namun, kebutuhan untuk mengubah gambar dua dimensi menjadi bentuk tiga dimensi semakin meningkat, terutama seiring dengan berkembangnya teknologi pencetakan *3D* dan manufaktur berbasis *digital* [2].

Konversi gambar vektor ke objek tiga dimensi menjadi tantangan tersendiri bagi banyak pengguna, terutama mereka yang tidak memiliki latar belakang teknis di bidang desain *3D*. Proses ini biasanya

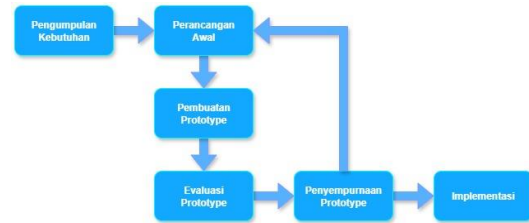
memerlukan perangkat lunak khusus yang tidak selalu mudah diakses atau digunakan oleh semua kalangan [3]. Selain itu, sebagian besar perangkat lunak konversi yang ada masih berbasis desktop, sehingga membatasi fleksibilitas dan aksesibilitas pengguna yang ingin melakukan konversi secara cepat dan praktis melalui perangkat apapun [4].

Melihat permasalahan tersebut, dibutuhkan solusi yang dapat memudahkan proses konversi gambar vektor ke objek tiga dimensi secara otomatis dan efisien. Solusi berbasis web menjadi pilihan yang tepat karena dapat diakses dari lokasi mana pun kapan saja tanpa memerlukan instalasi perangkat lunak lebih lanjut [5]. Dengan adanya layanan konversi berbasis web, diharapkan pengguna dengan latar belakang yang berbeda dapat dengan mudah pengguna dari berbagai latar belakang dapat dengan mudah mengubah gambar *SVG* menjadi objek *3D* untuk berbagai keperluan, seperti pembuatan *prototype*, desain produk, maupun kebutuhan edukasi [6].

B. METODE PENELITIAN

1. Metode Prototype

Metode penelitian *prototype* merupakan pendekatan pengembangan sistem yang digunakan, memungkinkan pembuatan prototipe awal yang dievaluasi dan ditingkatkan berdasarkan masukan pengguna. Enam langkah yang membentuk metode penelitian *prototype*: Pengumpulan kebutuhan, perancangan awal, pembuatan *prototype*, evaluasi *prototype*, penyempurnaan *prototype*, dan implementasi.



Gambar 1. Metode Prototype

Tahapan-tahapan berikut dari siklus hidup pengembangan perangkat lunak dicakup oleh teknik *prototype*, yang merupakan pendekatan penelitian yang digunakan:

- a) Pengumpulan Kebutuhan : Mengidentifikasi kebutuhan pengguna, seperti kemampuan mengunggah file, melakukan konversi, menampilkan pratinjau hasil.
- b) Perancangan Awal : Membuat rancangan awal tampilan antarmuka website, termasuk penempatan fitur-fitur yang dibutuhkan.
- c) Pembuatan *Prototype* : Mengembangkan *prototype* website yang sudah dapat menerima file *SVG*, menampilkan pratinjau *SVG*, dan melakukan konversi sederhana ke objek *3D*.
- d) Evaluasi *Prototype* : Melakukan pengujian *prototype* dengan melibatkan pengguna untuk mendapatkan masukan terkait kemudahan penggunaan, tampilan, dan hasil konversi.
- e) Penyempurnaan *Prototype* : Melakukan perbaikan pada fitur konversi, penambahan opsi pengaturan ketebalan objek *3D*, serta peningkatan tampilan dan interaksi pengguna.
- f) Implementasi : Mengembangkan website secara penuh, menambahkan fitur ekspor ke

format *STL*, melakukan pengujian akhir, dan memastikan website dapat digunakan secara optimal oleh pengguna.

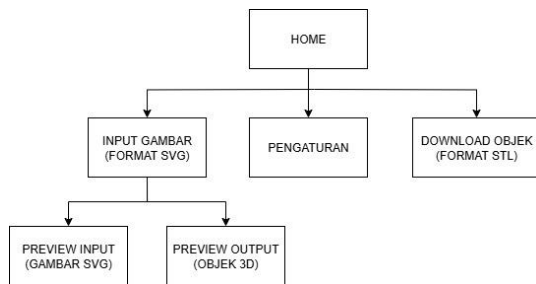
2. Kebutuhan Peralatan

Peralatan yang dibutuhkan dalam penelitian ini mencakup beberapa peralatan yang berbeda, seperti Bahasa pemrograman, *framework*, software pengolah *3D* dan *hosting*. Kebutuhan peralatan ini diperlukan guna mempermudah pengembangan sistem pada penelitian ini.

- a) Bahasa pemrograman yang digunakan adalah bahasa pemrograman *Javascript*, yang dapat memberikan logika dan interaktivitas dinamis pada halaman web. Bersama dengan bahasa pemrograman *HTML* dan *CSS* yang digunakan untuk membuat kerangka tampilan website dan beberapa fitur interaktif seperti desain dan modifikasi elemen halaman.
- b) *Hosting*. *Github Pages* adalah layanan hosting statis yang disediakan oleh *Github* untuk menampilkan halaman web langsung dari *repository Github*. Dengan *Github Pages*, pengembang dapat langsung mempublikasikan website, dokumentasi, atau portofolio secara gratis hanya dengan mengelola kode sumber di *repository*. Banyak teknologi web, termasuk *HTML*, *CSS*, dan *Javascript*, didukung oleh layanan ini. serta integrasi dengan generator situs statis seperti *Jekyll*. *Github Pages* memudahkan tahapan proses *deployment* karena setiap perubahan pada *repository* dapat langsung diupdate ke website secara otomatis [7].
- c) *Framework* pada Perancangan website ini menggunakan *framework Vite* yang dapat berfungsi agar website dapat memproses dengan cepat dan mengimpor file-file serta modul-modul yang diperlukan dengan mudah [8]. Pada bagian untuk memproses konversi, website ini menggunakan bahasa pemrograman salah satu *library* yang tersedia pada bahasa pemrograman *Javascript*, yaitu *library Three js* yang akan digunakan untuk pembuatan sistem pada website ini, karena *Three js* dapat digunakan untuk pengembangan yang ada hubungannya dengan dunia 3 dimensi, salah satunya adalah melakukan manipulasi gambar yang memiliki format *SVG (Scalable Vector Graphics)* objek 3 dimensi.
- d) *Software Pengolah 3 Dimensi*. *Software* yang digunakan untuk mengolah objek 3 dimensi adalah *Blender*, *Blender* adalah program open source untuk memproses dan menghasilkan efek visual *3D*, animasi, dan gambar. *Blender* dapat digunakan untuk memodelkan objek 3 dimensi, membuat animasi, *rendering* gambar, hingga melakukan simulasi fisika seperti cairan dan asap. *Software* ini banyak digunakan oleh seniman *digital*, animator, hingga pengembang game karena fiturnya yang lengkap dan gratis untuk digunakan. Selain itu, *Blender* memiliki komunitas yang aktif, sehingga pengguna dapat dengan mudah menemukan tutorial, plugin, dan dukungan teknis.

3. Struktur Navigasi

Struktur navigasi hirarki digunakan karena memiliki banyak keuntungan penting, seperti mengurangi beban kognitif pengguna, dan dapat meningkatkan kemudahan pencarian. Struktur navigasi hierarki biasanya diperjelas oleh representasi visual dari struktur ini, yang seringkali berbentuk *sidebar*, menu *dropdown*, atau jalur *breadcrumb*.



Gambar 2. Struktur Navigasi Hirarki Website

4. Use Case Diagram

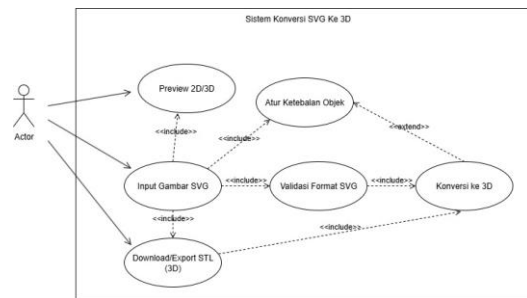
Use Case Diagram merupakan sebuah alat dalam pemodelan sistem berbasis *Unified Modeling Language (UML)* yang dapat menunjukkan bagaimana aktor (pengguna atau sistem eksternal) berinteraksi dengan sistem yang sedang dikembangkan.

Pengembang dapat lebih memahami persyaratan dan alur kerja bisnis dengan menggunakan diagram ini untuk mengidentifikasi fungsionalitas sistem dari sudut pandang pengguna. Komponen utama yang digunakan pada *activity diagram* ini adalah terdiri dari :

- Actor* : Sebagai peran pengguna yang berinteraksi dengan use case.
- Use Case* : Sebagai penggambaran dari suatu sistem yang terhubung dengan pengguna.
- Include* : Sebagai relasi pada *use case* tambahan yang mana

memerlukan *use case* lain untuk berfungsi.

- Extend* : Sebagai relasi pada *use case* tambahan yang mana dapat berdiri sendiri untuk berfungsi.
- Association* : Sebagai penghubung antara *actor* dengan *use case*.
- Generalization* : Sebagai identifikasi hubungan induk dan anak antara *use case*.



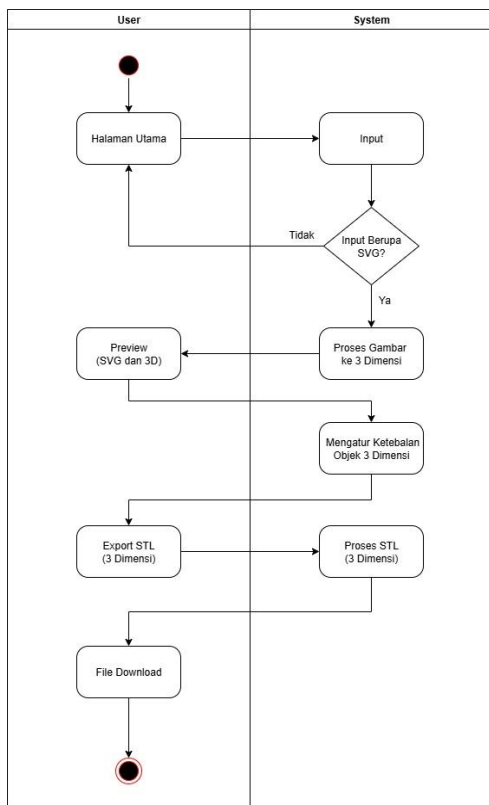
Gambar 3. Use Case Diagram

5. Activity Diagram

Dalam *Unified Modeling Language (UML)*, terdapat diagram yang merupakan diagram krusial yang digunakan untuk menggambarkan alur kerja suatu sistem atau proses bisnis, yaitu *activity diagram*. Dari awal hingga akhir, diagram ini menunjukkan urutan tindakan yang terjadi dalam sistem, seperti paralelisasi, sinkronisasi, dan pengambilan keputusan. *Activity Diagram* dalam *UML*, berbeda dengan diagram alur tradisional, lebih mampu merepresentasikan alur kerja yang rumit berkat fitur-fitur unik seperti *swimlane* untuk menetapkan peran kepada *aktor* dan *fork/join* untuk proses paralel. Komponen utama yang digunakan pada *activity diagram* ini adalah terdiri dari :

- Status Awal : Sebagai status dimulainya aliran kerja pada *activity diagram*, yang mana hanya terdapat satu status awal.

- b) Aktivitas : Sebagai aktivitas yang menyatakan aksi atau pekerjaan pada *activity diagram*.
- c) Percabangan : Sebagai menentukan keputusan, apakah program akan berjalan atau berhenti, ataupun mengulang ke status awal, biasanya ditentukan antara iya dan tidak.
- d) *Transition* : Sebagai penghubung aliran kerja antar aktivitas.
- e) Status Akhir : Sebagai status berakhirnya aliran kerja pada *activity diagram*, yang mana hanya terdapat satu status akhir.



Gambar 4. Activity Diagram

6. User Acceptance Testing

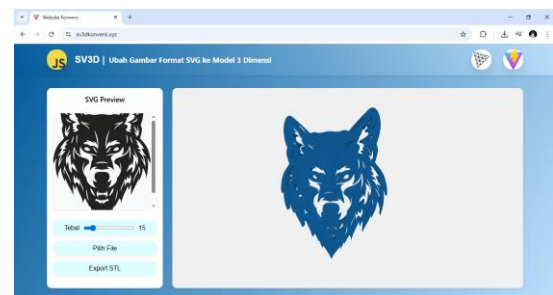
Langkah terakhir yaitu *User Acceptance Testing (UAT)*, memastikan sistem memenuhi kebutuhan pengguna akhir dan beroperasi sebagaimana mestinya. Untuk memastikan sistem

memenuhi kriteria penerimaan, pengguna atau perwakilan pengguna melakukan pengujian *UAT* sebelum peluncuran resmi sistem. Rumus skala *likert* berdasarkan respons responden—yang berkisar dari Sangat Buruk (STB), Buruk (TB), Cukup Baik (CB), Baik (B), hingga Sangat Baik (SB)—digunakan dalam studi ini untuk menghasilkan hasil *UAT*.

C. HASIL DAN PEMBAHASAN

1. Tampilan Website

Tampilan website yang dikembangkan menggunakan *HTML* dan *CSS*, tampilan website telah dibuat dengan se-simpel mungkin untuk mempermudah user melakukan navigasi pada tata letak halaman. Tata letak halaman website menampilkan dua panel, pada panel kiri menampilkan peralatan untuk pengguna, seperti melakukan input gambar berformatkan *SVG* dan download objek 3 dimensi. Pada panel kanan menampilkan output sementara hasil konversi gambar menjadi objek 3 dimensi.



Gambar 5. Tampilan Website

2. Import Pada Blender

Software Blender digunakan untuk melakukan *import* dari objek 3 dimensi yang telah di-download. Pada *software* pengolah 3 dimensi Blender, import dapat dilakukan dengan menekan tombol pada panel navigasi yang bertuliskan File, kemudian memilih tombol Import dan memilih format *STL*. Setelah itu, pilih file yang telah di-download untuk

memasukkan dan menampilkannya pada *software Blender*.

3. Hasil Pengujian *User Acceptance Testing*

Tabel 1. Hasil Pengujian *User Acceptance Testing*

No	Pertanyaan	S B	B	C B	T B	ST B
1.	Apakah tata letak dan tampilan website ini mudah dipahami?	5	11	4		
2.	Apakah navigasi antar menu pada website dapat dipahami?	6	12	2		
3.	Apakah anda merasa mudah dalam memahami cara kerja fitur konversi?	7	11	2		
4.	Apakah terdapat kendala dalam proses pengunggahan file SVG?	4	10	6		
5.	Apakah anda dapat melakukan penggunaan website dengan mudah?	7	10	3		
6.	Apakah terdapat kendala sistem pada website?	3	11	4	1	1
7.	Apakah fitur pada website sesuai dengan kebutuhan desainer 3D?	5	12	3		
8.	Apakah kualitas hasil konversi pada website ini baik dan memuaskan?	5	14	1		

9.	Apakah anda dapat memahami hasil pratinjau model 3D dengan mudah?	5	13	2		
10.	Apakah penggunaan website ini mempermudah pekerjaan desainer 3D?	9	9	2		

Dimulai dengan penilaian Sangat Buruk (STB) yang mendapat total 1 tanggapan, diikuti oleh Tidak Baik (TB) yang mendapat total 1 tanggapan, Cukup Baik (CB) yang mendapat 29 tanggapan, Baik (B) yang mendapat 113 tanggapan, dan Sangat Baik (SB) yang mendapat 56 tanggapan, tabel di atas menampilkan hasil keseluruhan untuk setiap skala.

Prosedur perhitungan skala Likert berikut diterapkan untuk mendapatkan hasil penjumlahan dari skenario tersebut:

- 5 x Total Respons Sangat Baik (SB) sama dengan Sangat Baik (SB).
- Baik (B) sama dengan 4 kali Total Respons Baik (B).
- Rumus untuk Cukup Baik (CB) adalah 3 x Total Respons Cukup Baik (CB).
- Buruk (TB) sama dengan 2 kali Total Respons Buruk (TB).
- 1 x Semua Respons Sangat Buruk (STB).

Hasilnya, skor dihitung menggunakan rumus yang telah disebutkan sebelumnya:

- Pada skala Sangat Baik (SB), respons responden sama dengan $56 \times 5 = 280$.
- Respons responden pada skala Baik (B) adalah $113 \times 4 = 452$.
- Respons responden pada skala Cukup Baik (CB) adalah $29 \times 3 = 87$.

- Respons responden pada skala Kurang Baik (TB) adalah $1 \times 2 = 2$.
- Pada skala Sangat Kurang (STB), respons responden adalah $1 \times 1 = 1$.
- Jumlah respons menghasilkan skor $280 + 452 + 87 + 2 + 1 = 822$.

Setelah hasil skor didapatkan, langkah selanjutnya adalah mendapatkan hasil interpretasi dari perhitungan skor. Untuk mendapatkan hasil interpretasi terlebih dahulu mengetahui skor tertinggi yaitu (Y) dan skor terendah yaitu (X), dengan penilain rumus seperti berikut :

- $Y = \text{skor tertinggi } likert \times \text{jumlah responden} \times \text{jumlah pertanyaan}$
- $X = \text{skor terendah } likert \times \text{jumlah responden} \times \text{jumlah pertanyaan}$

Jumlah skor :

- Sangat Baik (SB) = $5 \times (20 \times 10)$ (jumlah responden \times jumlah pertanyaan) = 1000
- Sangat Tidak Baik (STB) = $1 \times (20 \times 10)$ (jumlah responden \times jumlah pertanyaan) = 200

Setelah hasil interpretasi didapatkan, langkah selanjutnya adalah menentukan persentase *UAT*, yang digunakan rumus berikut :

$$\text{Hasil Persentase (\%)} = \frac{\text{Total skor}}{Y} \times 100\%$$

Dengan hasil :

$$\text{Hasil Persentase (\%)} = \frac{822}{1000} \times 100\% = 82,2\%$$

Hasil persentase akhir dari *user acceptance testing* adalah 82,2% yang berdasarkan dari hasil 10 pertanyaan dari 20 responden, dan dengan skor responden 822 skor. Dengan hasil persentase ini diketahui bahwa website beserta sistem yang telah dikembangkan telah bekerja dengan baik walaupun dengan sedikit

kendala yang ditemui oleh para pengguna.

D. KESIMPULAN DAN SARAN

1. Kesimpulan

Dengan berhasilnya pembuatan penelitian berupa website konversi gambar *SVG* menjadi objek 3 dimensi yang dibuat dengan menggunakan bahasa pemrograman *Javascript*, *HTML*, dan *CSS*, dan juga penggunaan *framework Vite* sebagai *module* dan *library Three js* sebagai *library* utama dalam pengembangan website ini. Website ini juga menggunakan penyedia layanan hosting gratis yang disediakan oleh *Github Pages*, sehingga pengembang tidak perlu mengeluarkan uang untuk membayar sebuah hosting untuk dapat mempublikasikan website konversi *SVG* ke *3D* ini. Pada hasil *User Acceptance Testing* juga diperoleh sebesar 82,2% yang mana pengguna dapat menggunakan website ini dengan baik melalui penilaian tata letak, antarmuka, sistem, dan kemudahan dalam membantu pekerjaan desainer 3 dimensi.

Website dapat berjalan dengan menerima input yang kemudian diproses oleh sistem hingga menampilkan ketiga *output*, yaitu *output preview*, *download*, dan *import* pada *software* pemroses *object* 3 dimensi, seperti contoh *software Blender* untuk pekerjaan pada bidang desain *3D*, terutama untuk pencetakan *3D*. Website konversi ini dapat diakses secara public dengan mengunjungi laman <https://sv3dkonversi.xyz/> pada browser masing-masing pengguna.

2. Saran

Saran yang diberikan adalah penambahan fitur-fitur yang dapat membuat dan menyimpan informasi pengguna seperti *Register* dan *Login*,

sehingga pengguna dapat juga menyimpan informasi konversi mereka dalam bentuk *history*. Penggunaan fitur *history* ini diharapkan membantu pengguna dalam kemudahan melakukan operasi pada website, sehingga pengguna tidak perlu lagi memasukkan input kepada website dan melakukannya berulang kali untuk melakukan konversi gambar ke model 3 dimensi yang berbeda.

E. REFERENSI

- [1] Lopes, R. G., Ha, D., Eck, D., & Shlens, J. (2019). A learned representation for scalable vector graphics. *Deep Generative Models for Highly Structured Data*, DGS@ICLR 2019 Workshop, 7930–7939.
- [2] Jisheng, L. (2015). *SVG 3D Graphical Presentation for Web-based Applications*. <https://pdfs.semanticscholar.org/84e5/77882ae54561674880eb2ddd8a90e52bceae.pdf>
- [3] Wibowo, M. C. (2022). Pemodelan Dengan Blender 3D. *Penerbit Yayasan Prima Agus Teknik*, 8(1), 1–505. <https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/332>
- [4] Utomo, P., & Falahah. (2020). Building Serverless Website on GitHub Pages. *IOP Conference Series: Materials Science and Engineering*, 879(1), 0–7. <https://doi.org/10.1088/1757-899X/879/1/012077>
- [5] Bose, S. (2022). *Rendering interactive 3D scene as a web-page using Three.js, WebGL, and Blender*. May.
- [6] Rosnelly, R., Wahyuni, L., Melvy Anggraini, G., & Lazuli, I. (2023). Implementasi Javascript Dalam Pembuatan Web Sederhana. *Community Service Journal) e-ISSN*, 2(1), 116–123. <https://www.doi.org/10.22303/coral.2.1.2023.116-123>
- [7] Gayatri Makrand Ghodke, & Trupti Chavan. (2024). An Overview of Git. *International Journal of Scientific Research in Modern Science and Technology*, 3(6), 17–23. <https://doi.org/10.59828/ijrmst.v3i6.216>
- [8] Nguyen, T. A. (2024). *A comparative analysis of Webpack and Vite as build tools for JavaScript*. 32.