**PENGEMBANGAN WEB (TEORI)**

**LAPORAN EKSPERIMEN MENGENAI PERBANDINGAN GREENSOCK(GSAP) DAN ANIMEJS**

*Laporan ini disusun untuk memenuhi tugas 4 mata kuliah Pengembangan Web (Teori)*



Disusun oleh kelompok B4:

Asri Husnul Rosadi 221524035

Faris Abulkhoir 221524040

Mahardika Pratama 221524044

Muhamad Fahri Yuwan 221524047

Najib Alimudin Fajri 221524053

Septyana Agustina 221524058

Sarah 221524059

Dosen Pengampu:

Joe Lian Min, M.Eng.

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA**

**PROGRAM STUDI D4 TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2024**

# **DAFTAR ISI**

[**DAFTAR ISI** i](#_Toc177312039)

[**A.** **IDENTIFIKASI PROBLEM** 1](#_Toc177312040)

[**B.** **DESKRIPSI PROBLEM** 1](#_Toc177312041)

[**C.** **METODOLOGI EKSPERIMEN** 1](#_Toc177312042)

[**D.** **PELAKSANAAN EKSPERIMEN** 2](#_Toc177312043)

[**E.** **PELAKSANAAN EKSPERIMEN** 3](#_Toc177312044)

[**F.** **KESIMPULAN** 3](#_Toc177312045)

Link chatGPT : https://chatgpt.com/share/66e6ad42-69f8-8001-a7eb-02b351b53b26

# **IDENTIFIKASI PROBLEM**

Perbandingan antara dua library animasi JavaScript, GSAP (GreenSock Animation Platform) dan Anime.js, dalam konteks animasi kompleks pada aplikasi web. GSAP terkenal dengan performa tinggi dan fleksibilitasnya dalam animasi yang rumit, sedangkan Anime.js menawarkan kemudahan penggunaan dan sintaks yang lebih sederhana. Masalahnya adalah menentukan library mana yang lebih efektif dan efisien dalam situasi animasi kompleks.

# **DESKRIPSI PROBLEM**

1. **Penjelasan Masalah:**

* **GSAP**: Memungkinkan pembuatan animasi yang sangat kompleks dengan fitur-fitur seperti timeline, kontrol waktu yang mendetail, dan plugin tambahan. Ini ideal untuk proyek besar dan animasi yang memerlukan performa tinggi dan kontrol yang mendalam.
* **Anime.js**: Menyediakan solusi animasi yang lebih sederhana dan cepat dengan sintaks yang mudah dipahami. Cocok untuk animasi sederhana hingga menengah, tetapi mungkin kurang fleksibel untuk animasi yang sangat kompleks dan memerlukan kontrol yang mendalam.

**2.** **Dampak**:

* **GSAP**: Memberikan kontrol yang lebih baik atas animasi, memungkinkan pembuatan efek visual yang lebih canggih dan performa yang lebih halus untuk animasi kompleks. Namun, mungkin memerlukan waktu belajar yang lebih lama dan lebih banyak konfigurasi.
* **Anime.js**: Menawarkan kemudahan penggunaan dan sintaks yang lebih bersih, mempercepat pengembangan animasi. Namun, untuk animasi yang sangat kompleks, mungkin kurang optimal dibandingkan dengan GSAP dalam hal performa dan kontrol.

# **METODOLOGI EKSPERIMEN**

**1.** **Tujuan Eksperimen**: Menilai dan membandingkan efektivitas **GSAP** dan **Anime.js** dalam mengelola animasi kompleks dari perspektif fleksibilitas, performa, dan kemudahan penggunaan.

**2.** **Langkah-langkah Eksperimen**:

* **Implementasi dengan GSAP:** Membuat animasi kompleks yang melibatkan beberapa elemen dengan menggunakan GSAP. Menilai performa, kontrol, dan kemudahan implementasi animasi berurutan dan bersamaan.
* **Implementasi dengan Anime.js:** Membuat animasi yang sama dengan menggunakan Anime.js. Menilai kemudahan penggunaan, efektivitas dalam mengelola animasi kompleks, dan perbandingan performa.

**3.** **Alat dan Teknologi**: GSAP dan Anime.js untuk animasi, Browser Developer Tools untuk memonitor performa, Code Editor untuk pengembangan dan modifikasi kode.

# **PELAKSANAAN EKSPERIMEN**

|  |
| --- |
| **Animation Code** |
|  |

|  |
| --- |
| **Hasil** |
|  |

# **PELAKSANAAN EKSPERIMEN**

|  |
| --- |
| 1. Hasil Pengujian:   * GSAP: Menunjukkan kemampuan untuk menangani animasi kompleks dengan kontrol waktu yang mendetail dan performa yang sangat baik. Animasi berjalan dengan sangat halus meskipun melibatkan banyak elemen dan efek. * Anime.js: Menyediakan kemudahan implementasi dengan sintaks yang bersih. Animasi juga berjalan dengan baik, tetapi dalam kasus animasi yang sangat kompleks, beberapa keterbatasan dalam kontrol dan performa mungkin terlihat.   2. Evaluasi:   * GSAP lebih unggul dalam hal performa untuk animasi yang kompleks dan menawarkan kontrol yang lebih mendalam. * Anime.js lebih mudah digunakan dan ideal untuk proyek yang memerlukan animasi sederhana hingga menengah dengan performa yang memadai.   3. Kendala:   * GSAP: Memiliki kurva pembelajaran yang lebih tinggi dan mungkin memerlukan lebih banyak konfigurasi. * Anime.js: Meskipun lebih sederhana, mungkin tidak menangani animasi sangat kompleks sebaik GSAP dan dapat menunjukkan batasan dalam pengaturan kontrol yang sangat rinci. |

# **KESIMPULAN**

* 1. Ringkasan: Eksperimen menunjukkan bahwa GSAP memberikan keuntungan signifikan dalam hal performa dan kontrol untuk animasi yang sangat kompleks. Anime.js menawarkan kemudahan penggunaan yang lebih besar dan cukup efektif untuk animasi sederhana hingga menengah.
  2. Rekomendasi: Disarankan untuk menggunakan GSAP jika proyek memerlukan animasi yang sangat kompleks dan performa yang tinggi. Anime.js lebih cocok untuk proyek dengan animasi yang lebih sederhana dan ketika kemudahan penggunaan dan implementasi cepat lebih diutamakan.
  3. Langkah Berikutnya: Implementasikan GSAP untuk proyek animasi yang kompleks dan pantau performa serta responsivitas. Pertimbangkan Anime.js untuk proyek dengan kebutuhan animasi yang lebih sederhana, dan terus evaluasi kedua library berdasarkan kebutuhan spesifik proyek.