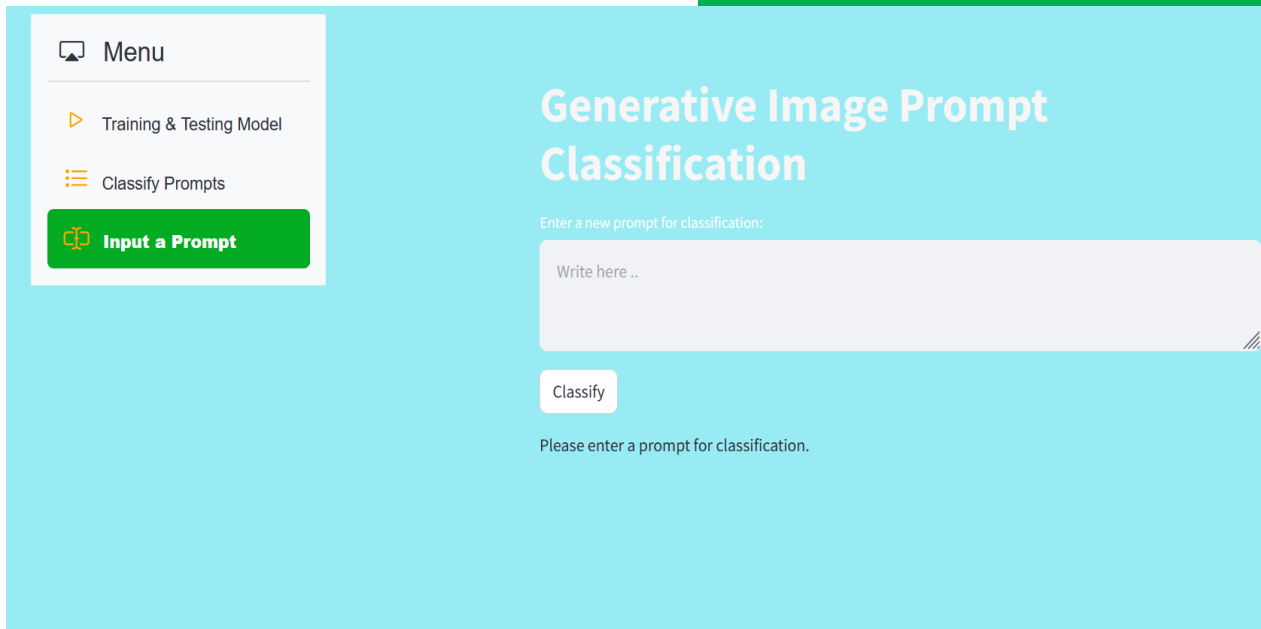


# LAPORAN PENGUJIAN

## Klasifikasi Image Prompt Generative pada Website



Menu

- Training & Testing Model
- Classify Prompts
- Input a Prompt**

### Generative Image Prompt Classification

Enter a new prompt for classification:

Write here ..

Classify

Please enter a prompt for classification.

### Disusun Oleh:

Samuel P Tambunan (3312211055)  
Sechan Faradila Sahab (3312211004)  
Muhamad Rafiansyah (3312211006)  
Putra Ramadhan (3312211008)  
Alpaber P Tambunan (3312211073)

**Jurusan Teknik Informatika**  
**Politeknik Negeri Batam**

**HALAMAN PENGESAHAN**  
**LAPORAN PENGUJIAN PERANGKAT LUNAK**

"Klasifikasi Image Prompt Generative pada Website Promptails"

**Ketua Proyek,**

Samuel Parsaoran Tambunan  
NIM. 3312211055

**Anggota 2,**

Muhamad Rafiansyah  
NIM. 3312211006

**Anggota 4,**

Alpaber Pardomuan Tambunan  
NIM. 3312211073

**Anggota 1,**

Sechan Faradila Sahab  
NIM. 3312211004

**Anggota 3,**

Putra Ramadhan  
NIM. 3312211008

**Manajer Proyek,**

Agung Riyadi, S.Si. M.Kom  
NIP. 119221

**Menyetujui,**

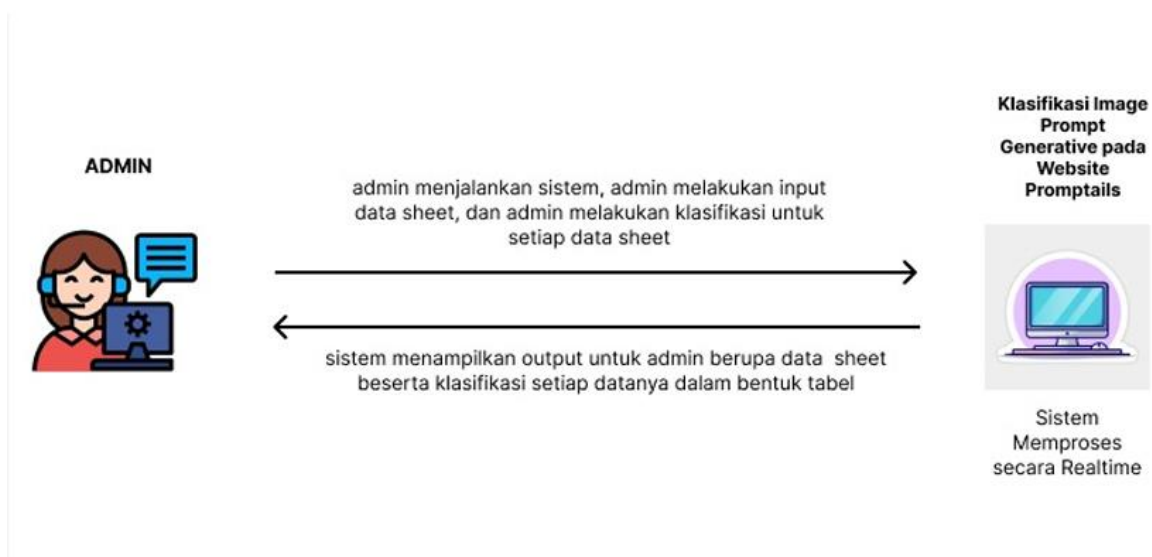
Agung Riyadi, S.Si. M.Kom  
NIP. 119221

## 1. Executive Summary

### A. Deskripsi Umum

Situs *web Promptails* memiliki lebih dari 30 juta data *prompt* hasil kecerdasan buatan generatif. Dari data ini, sebagian sudah dapat di klasifikasikan berdasarkan beberapa kategori berdasarkan gaya seperti *isometric*, *gravity*, *abstract*, *fractal*, *oil paintings* dan lainnya. Namun, masih banyak yang belum terklasifikasikan. Oleh karena itu, dengan menggunakan *data mining*, kami belajar membuat model dari data yang telah diidentifikasi agar data yang belum terklasifikasikan dapat diklasifikasikan dengan baik. Kami juga menghitung *error* untuk hasil klasifikasi ini.

Berikut adalah gambaran umum dari sistem yang akan kami bangun :



Berdasarkan gambar diatas, fitur yang dapat dijalankan oleh admin terhadap sistem adalah :

- Admin dapat melakukan input *data sheet*
- Admin dapat melakukan klasifikasi untuk setiap *data sheet* setelah menjalankan sistem.

Sedangkan sistem klasifikasi image prompt generatif pada *web promptails* terhadap admin mengeluarkan output berupa data sheet beserta klasifikasi setiap data berupa gambar.

## B. User Stories

User Stories adalah deskripsi singkat yang menjelaskan kebutuhan pengguna dalam pengembangan perangkat lunak. Alat ini membantu kita memahami apa yang diinginkan pengguna dari sistem atau produk yang sedang kita kembangkan.

Judul : Klasifikasi <i>Image Prompt Generative</i> pada <i>Website Promptails</i>	
Akses Pengguna	Akses
Pengguna	Sebagai pengguna, saya ingin melihat hasil klasifikasi gambar-gambar baru pada <i>website Promptails</i> .
Administrator	Sebagai administrator, saya ingin dapat mengklasifikasikan <i>prompt</i> hasil dari <i>AI-based Image Generative Prompt</i> sesuai dengan klasifikasi yang telah ditentukan.

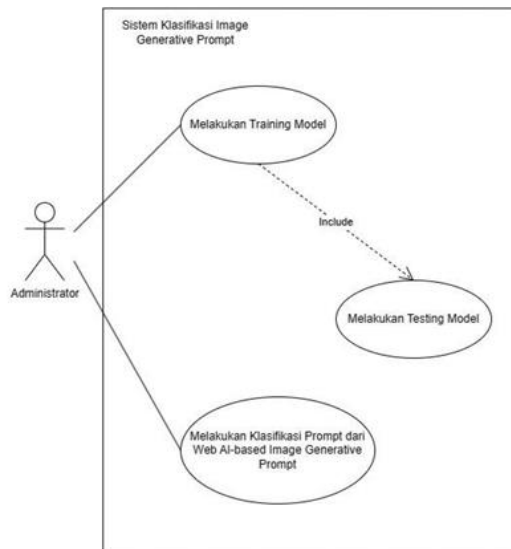
## C. Pemodelan Sistem

- Use Case Diagram

Use case diagram adalah salah satu jenis diagram UML (Unified Modelling Language) yang menggambarkan hubungan interaksi antara sistem dan aktor. Use Case dapat mendeskripsikan tipe interaksi antara pengguna sistem dengan sistemnya.

Berikut adalah tujuan pembuatan use case:

- Menunjukkan proses aktivitas secara urut dalam sistem.
- Menggambarkan proses bisnis, bahkan menampilkan urutan aktivitas pada sebuah proses.
- Menjadi jembatan antara pembuat dengan konsumen untuk mendeskripsikan sebuah sistem. Berikut adalah Use case dari sistem perangkat lunak website aplikasi Klasifikasi Image Prompt Generative pada Website Promptails yang akan dibangun:



Gambar 6

Pada gambar 6, dapat di simpulkan bahwa sistem yang akan dibangun memiliki satu aktor yaitu admin dan dua usecase atau interaksi antar admin dan sistem. Admin dapat melakukan training serta testing model, admin juga dapat melakukan Klasifikasi Prompt dari Web AI based Image Generative Prompt.

#### D. Aktor-Aktor yang Terlibat

Beberapa aktor utama yang terlibat dalam proyek ini adalah:

- Pemilik Proyek: Promptails, diwakili oleh Agung Riyadi, S.Si. M.Kom, yang juga bertindak sebagai manajer proyek.
- Tim Pengembang: Tim ini terdiri dari Samuel Parsaoran Tambunan, Sechan Faradila Sahab, Muhamad Rafiansyah, Putra Ramadhan, dan Alpaber Pardomuan Tambunan. Mereka bertanggung jawab atas desain, pengembangan, dan pengujian aplikasi.
- Pengguna Akhir: Kreator konten, desainer grafis, dan pengguna umum yang menggunakan aplikasi untuk menghasilkan dan mengelola prompt gambar AI.

## E. Jenis Platform Aplikasi

Aplikasi ini adalah platform berbasis desktop App. Dengan desain yang responsif, aplikasi ini memberikan pengalaman pengguna yang konsisten dan optimal di berbagai resolusi layar.

## F. Alasan Pelaksanaan Pengujian pada Aplikasi PBL

Pengujian aplikasi merupakan langkah penting dalam siklus pengembangan perangkat lunak untuk memastikan bahwa aplikasi berfungsi sebagaimana mestinya dan memenuhi kebutuhan pengguna. Berikut adalah beberapa alasan utama pelaksanaan pengujian pada aplikasi PBL:

- **Memastikan Kualitas dan Kinerja:** Pengujian membantu mengidentifikasi dan memperbaiki bug atau kesalahan dalam aplikasi sebelum dirilis ke pengguna. Hal ini memastikan bahwa aplikasi berfungsi dengan baik dan memberikan pengalaman pengguna yang mulus.
- **Validasi Fitur dan Fungsi:** Melalui pengujian, tim dapat memverifikasi bahwa semua fitur dan fungsi yang direncanakan bekerja sesuai dengan spesifikasi. Ini mencakup pengujian unit, integrasi, dan sistem.
- **Kompatibilitas:** Pengujian dilakukan untuk memastikan bahwa aplikasi berjalan dengan baik di berbagai perangkat dan platform, menghindari masalah kompatibilitas.
- **Kepuasan Pengguna:** Dengan melakukan pengujian yang komprehensif, tim pengembang dapat memastikan bahwa aplikasi memenuhi ekspektasi dan kebutuhan pengguna, yang pada akhirnya meningkatkan kepuasan pengguna dan kepercayaan pada aplikasi.

## 2. Standar Prioritas Pengujian

Perhatikan tabel standar prioritas pengujian berikut ini:

Prioritas	Keterangan
LOW	Fungsional yang memiliki dampak minimal bagi pengguna akhir dan tidak mempengaruhi proses bisnis utama aplikasi.
MEDIUM	Fungsionalitas penting tetapi tidak mendesak. Dapat ditunda jika diperlukan
HIGH	Fungsionalitas utama aplikasi yang memiliki dampak besar terhadap pengguna akhir ataupun proses bisnis.

**Prioritas LOW:**

- UI/UX Enhancements: Optimasi tampilan visual dan pengalaman pengguna yang tidak mempengaruhi fungsi utama aplikasi.
- Informational Messages: Pengujian pesan-pesan informasi yang tidak esensial untuk operasi sehari-hari aplikasi.

**Prioritas MEDIUM:**

- Reporting Features: Pengujian fitur laporan dan analisis yang bermanfaat tetapi tidak mendesak.
- User Personalization: Pengujian fitur personalisasi pengguna yang dapat meningkatkan pengalaman tetapi tidak kritis.

**Prioritas HIGH:**

- Classification Core Functionality: Pengujian fitur utama yang memungkinkan klasifikasi gambar oleh admin. Ini adalah fungsionalitas inti dari aplikasi.
- Security and Authentication: Pengujian mekanisme keamanan dan autentikasi untuk melindungi data pengguna.
- Performance and Scalability: Pengujian untuk memastikan aplikasi dapat menangani beban pengguna yang tinggi dan tetap berjalan lancar

### 3. Kebutuhan Fungsional Aplikasi

Berikut adalah fitur-fitur fungsional yang ada pada aplikasi PBL, beserta prioritas dari masing-masing fungsionalitasnya:

REF ID	Nama Fungsi	Aktor	Prioritas
F001	Admin dapat memasukkan dataset ke dalam model	Admin	High
F002	Admin dapat melakukan training dataset pada model	Admin	High
F003	Admin dapat melakukan testing dataset pada model	Admin	High
F004	Admin dapat melakukan klasifikasi prompt dari AI-based Image Generative Prompt	Admin	High
F005	Admin dapat menyimpan hasil klasifikasi prompt ke dalam database	Admin	Medium

#### 4. Kebutuhan Non Fungsional Aplikasi

*Sebutkan fitur non fungsional apa saja yang ada pada aplikasi PBL.*

REF ID	Non Fungsional
NF001	Waktu Muat Halaman: Halaman harus dimuat dalam waktu kurang dari 15 detik.
NF002	Keamanan: Sistem harus memiliki lapisan keamanan yang memadai untuk melindungi data sensitif.
NF003	Skalabilitas: Sistem harus dapat meningkatkan kapasitasnya secara horizontal atau vertikal ketika diperlukan.
NF004	Ketersediaan: Sistem harus tersedia hampir 24 jam setiap hari untuk pengguna.
NF005	Kepatuhan Hukum dan Regulasi: Sistem harus mematuhi hukum dan regulasi terkait privasi, keamanan, dan aksesibilitas sesuai dengan UU No.27 Tahun 2022 tentang Perlindungan Data Pribadi.

#### 5. Hasil User Acceptance Testing

User Acceptance Testing (UAT) merupakan tahap kritis dalam pengembangan aplikasi untuk memastikan bahwa sistem memenuhi kebutuhan dan harapan pengguna akhir. UAT dilakukan untuk memverifikasi bahwa semua fitur berfungsi dengan baik dan aplikasi siap digunakan dalam lingkungan nyata.

##### A. Mengapa UAT Dilakukan

UAT dilakukan untuk memastikan bahwa aplikasi memenuhi persyaratan fungsional dan non-fungsional yang telah ditetapkan. Ini juga bertujuan untuk mengidentifikasi dan memperbaiki masalah yang mungkin terjadi sebelum aplikasi diimplementasikan secara luas. Dengan melibatkan pengguna akhir, UAT membantu memastikan bahwa aplikasi benar-benar sesuai dengan kebutuhan mereka dan siap digunakan dalam lingkungan operasional sebenarnya.

##### B. Fitur yang Diuji

Dalam UAT ini, beberapa fitur utama yang diuji antara lain:

- F001: Input Data Sheet - Admin dapat memasukkan dataset ke dalam model.
- F002: Training Dataset - Admin dapat melakukan pelatihan dataset pada model.
- F003: Testing Dataset - Admin dapat melakukan pengujian dataset pada model.
- F004: Klasifikasi Prompt - Admin dapat mengklasifikasikan prompt dari AI-based Image Generative Prompt.
- F005: Penyimpanan Hasil Klasifikasi - Admin dapat menyimpan hasil klasifikasi prompt ke dalam database.



### **C. Aktor yang Terlibat**

Aktor yang terlibat dalam UAT ini meliputi:

- Irgi
- Meysy
- David
- Galih
- Tim PBL IF 23-2-16

### **D. Kendala yang Dihadapi**

Selama pelaksanaan UAT, beberapa kendala yang dihadapi antara lain:

- Teknis: Beberapa fitur mengalami bug yang menghambat proses pengujian, seperti masalah kompatibilitas dan performa aplikasi.
- Waktu: Terbatasnya waktu untuk melakukan pengujian menyeluruh pada semua fitur aplikasi.

### **E. Statistik UAT**

- Jumlah Test Scenario: 5 skenario
- Jumlah Berhasil: 5 skenario
- Jumlah Gagal: 0 skenario

## **5. Hasil Load Testing**

Load Testing adalah bagian penting dari pengujian non-fungsional yang dilakukan untuk memastikan bahwa aplikasi dapat menangani beban yang diharapkan dalam lingkungan produksi. Berikut adalah rangkuman dari hasil load testing yang telah dilakukan.

### **A. Mengapa Load Testing Dilakukan**

Load testing dilakukan untuk mengevaluasi kinerja aplikasi di bawah beban kerja yang bervariasi. Tujuannya adalah untuk memastikan bahwa aplikasi tetap responsif, stabil, dan dapat diandalkan ketika digunakan oleh sejumlah besar pengguna secara bersamaan. Pengujian ini membantu dalam mengidentifikasi batas kapasitas aplikasi dan mengungkap potensi bottleneck atau masalah kinerja yang perlu diperbaiki.

## **B. Fitur yang Diuji**

Beberapa fitur utama yang diuji selama load testing meliputi:

- Input Data Sheet: Menguji kemampuan aplikasi untuk menerima dan memproses input data dalam jumlah besar secara bersamaan.
- Training Dataset: Mengukur kinerja aplikasi saat melakukan pelatihan dataset dengan banyak data.
- Testing Dataset: Evaluasi performa aplikasi ketika menjalankan pengujian dataset yang besar.
- Klasifikasi Prompt: Menguji kecepatan dan akurasi aplikasi dalam mengklasifikasikan prompt AI-based image generative.
- Penyimpanan Hasil Klasifikasi: Menilai kemampuan aplikasi dalam menyimpan hasil klasifikasi ke database di bawah beban tinggi.

## **C. Jumlah Virtual User yang Digunakan**

Pengujian dilakukan dengan menggunakan 6 virtual users yang secara bersamaan mengakses dan menggunakan fitur-fitur utama aplikasi. Hal ini untuk mensimulasikan skenario penggunaan di dunia nyata dengan beban kerja yang tinggi.

## **D. Kendala yang Dihadapi**

Selama pelaksanaan load testing, beberapa kendala yang dihadapi antara lain:

- Sumber Daya Server: Terjadi keterbatasan pada sumber daya server yang menyebabkan beberapa tes tidak berjalan optimal.
- Waktu Respon yang Tinggi: Pada beban puncak, waktu respon aplikasi meningkat drastis, menyebabkan delay yang signifikan.
- Error Rate: Ada peningkatan error rate yang menunjukkan bahwa aplikasi gagal menangani beberapa permintaan pada beban tinggi.

## **E. Statistik Aplikasi**

Hasil load testing menunjukkan statistik berikut ini:

- Response Time: Rata-rata waktu respon adalah 30 detik pada beban puncak, dengan puncak waktu respon mencapai 60 detik.

- Error Rate: Terdapat 5% error rate, menunjukkan bahwa 25 dari 500 permintaan mengalami kegagalan.

## 1. Tujuan Pengujian

Tujuan dari load testing ini adalah untuk mengukur dan mengevaluasi kinerja aplikasi PBL di bawah beban pengguna yang tinggi, memastikan aplikasi tetap responsif, dan menentukan batas maksimum beban yang dapat ditangani oleh aplikasi.

## 2. Lingkup Pengujian

Pengujian mencakup semua fitur utama aplikasi yang digunakan oleh admin, termasuk:

- Pengunggahan dataset ke dalam model
- Pelatihan model dengan dataset
- Pengujian model dengan dataset
- Klasifikasi prompt dari AI-based Image Generative Prompt
- Penyimpanan hasil klasifikasi ke dalam database

## 3. Kriteria Keberhasilan

- > Aplikasi dapat menangani hingga 50 pengguna virtual secara bersamaan tanpa kegagalan sistem yang signifikan.
- > Waktu respon rata-rata untuk semua permintaan di bawah 60 detik.
- > Error rate tidak melebihi 5% selama pengujian.
- > Sistem tidak mengalami crash atau penurunan kinerja yang signifikan.

## 4. Metodologi Pengujian

- > Durasi Pengujian: Setiap skenario akan dijalankan selama 1 jam.

## 6. Kesimpulan

### Ringkasan Temuan

- Kualitas dan Kinerja: Pengujian menunjukkan bahwa aplikasi berfungsi dengan baik dengan beberapa perbaikan yang diperlukan untuk optimalisasi kinerja.
- Validasi Fitur dan Fungsi: Semua fitur utama berhasil diuji dan bekerja sesuai dengan spesifikasi, namun beberapa peningkatan pada fitur minor dapat dipertimbangkan.

- Kompatibilitas: Aplikasi menunjukkan kompatibilitas yang baik di berbagai perangkat dan platform.
- Load Testing: Aplikasi dapat menangani beban tinggi dengan respons waktu yang memadai, meskipun terdapat beberapa peningkatan yang diperlukan untuk mengurangi error rate dan waktu respon.

## **Rekomendasi**

- Optimisasi Kinerja: Lakukan optimisasi pada server dan aplikasi untuk mengurangi waktu respon dan meningkatkan kinerja pada beban tinggi.
- Bug Fixing: Identifikasi dan perbaiki bug yang ditemukan selama pengujian untuk meningkatkan stabilitas dan keandalan aplikasi.
- Pengujian Lanjutan: Lakukan pengujian lanjutan setelah perbaikan untuk memastikan bahwa semua peningkatan telah berhasil diimplementasikan dan aplikasi siap untuk digunakan secara luas.

## **Implikasi Terhadap Proyek**

Hasil pengujian menunjukkan bahwa aplikasi memiliki potensi besar untuk memenuhi kebutuhan pengguna dengan beberapa peningkatan. Pengujian lanjutan dan optimisasi akan memastikan bahwa aplikasi dapat memberikan pengalaman pengguna yang optimal dan dapat diandalkan dalam jangka panjang. Proyek ini menunjukkan kemajuan yang signifikan dan dengan perbaikan yang tepat, aplikasi ini siap untuk diluncurkan kepada pengguna akhir.

**LAMPIRAN A. DAFTAR HADIR PELAKSANAAN PENGUJIAN****DAFTAR HADIR PELAKSANAAN PENGUJIAN**

**Agenda** : Pengujian UAT aplikasi X  
**Lokasi Kegiatan** : Online Zoom  
**Tanggal Pelaksanaan** : Minggu, 2 Juni 2024  
**Waktu Pelaksanaan** : 09.00 s/d 12.00 WIB

**DAFTAR HADIR PESERTA**

No	Nama	Peran	No. Kontak (hp/email)	Tanda Tangan
1	Irgi Haslan Putra	Rekan Kelas		
2	David Patrick Ken	Rekan Kelas		
3	Meysy Sinaga	Rekan Jurusan		
4	Galih Ivan Aditya	Rekan Kelas		

## LAMPIRAN B. TEST PLAN & EXECUTION USER ACCEPTANCE TEST

### TEST PLAN & EXECUTION USER ACCEPTANCE TEST

**Nama Proyek** : PBL IF 23-2-16  
**Judul PBL** : Klasifikasi Image Prompt Generative pada Website  
**Manajer Proyek** : Agung Riyadi, S.Si. M.Kom  
**Tanggal Dokumen** : 9 Juni 2024

### HASIL UJI USER ACCEPTANCE TESTING (UAT)


UAT ID	Skenario Pengujian	Nama Penguji	Status	Komentar Tambahan
UAT01	<p><b>ID Referensi Fungsional:</b> F001</p> <p><b>Deskripsi Pengujian:</b> Pengujian ini dilakukan untuk memastikan bahwa admin dapat memasukkan dataset ke dalam model dengan benar. Tujuan pengujian adalah untuk memverifikasi bahwa sistem dapat menerima dan memproses dataset yang diberikan oleh admin tanpa error.</p> <p><b>Skenario Pengujian:</b></p> <ol style="list-style-type: none"><li>1. Buka browser</li><li>2. Navigasi ke halaman <i>Input a Prompt</i></li><li>3. Upload file dataset</li><li>4. Klik tombol submit</li></ol> <p><b>Hasil yang diharapkan:</b> User diarahkan ke halaman konfirmasi dengan pesan "Dataset berhasil diunggah."</p>	Irgi Haslan Putra	Berhasil	-
UAT02	<p><b>ID Referensi Fungsional:</b> F002</p> <p><b>Deskripsi Pengujian:</b> Pengujian ini dilakukan untuk memastikan bahwa admin dapat melakukan training dataset pada model dengan benar. Tujuan pengujian adalah untuk memverifikasi bahwa proses training berjalan sesuai dengan spesifikasi tanpa adanya error atau gangguan.</p> <p><b>Skenario Pengujian:</b></p> <ol style="list-style-type: none"><li>1. Buka browser</li><li>2. Navigasi ke halaman <i>Training &amp; Testing Model</i></li><li>3. Pilih dataset yang ingin di-train</li><li>4. Klik tombol "Start Training"</li><li>5. Tunggu hingga proses training selesai</li></ol>	Meysy Sinaga	Berhasil	-

	<b>Hasil yang diharapkan:</b> Sistem menampilkan pesan "Training berhasil" dan hasil model yang di-train.			
UAT03	<b>ID Referensi Fungsional:</b> F003  <b>Deskripsi Pengujian:</b> Pengujian ini dilakukan untuk memastikan bahwa admin dapat melakukan testing dataset pada model dengan benar. Tujuan pengujian adalah untuk memverifikasi bahwa proses testing berjalan sesuai dengan spesifikasi dan hasilnya akurat.  <b>Skenario Pengujian:</b> <ol style="list-style-type: none"> <li>1. Buka browser</li> <li>2. Navigasi ke halaman <i>Training &amp; Testing Model</i></li> <li>3. Pilih dataset yang ingin di-test</li> <li>4. Klik tombol "Start Testing"</li> <li>5. Tunggu hingga proses testing selesai</li> </ol> <b>Hasil yang diharapkan:</b> Sistem menampilkan hasil testing dengan statistik kinerja model.	David Patrick Ken	Berhasil	-
UAT04	<b>ID Referensi Fungsional:</b> F004  <b>Deskripsi Pengujian:</b> Pengujian ini dilakukan untuk memastikan bahwa admin dapat mengklasifikasikan prompt dari AI-based Image Generative Prompt dengan benar. Tujuan pengujian adalah untuk memverifikasi bahwa klasifikasi berjalan dengan baik dan hasilnya sesuai dengan yang diharapkan.  <b>Skenario Pengujian:</b> <ol style="list-style-type: none"> <li>1. Buka browser</li> <li>2. Navigasi ke halaman <i>Classify Prompts</i></li> <li>3. Pilih prompt yang ingin diklasifikasikan</li> <li>4. Klik tombol "Start Classification"</li> <li>5. Tunggu hingga proses klasifikasi selesai</li> </ol> <b>Hasil yang diharapkan:</b> Sistem menampilkan hasil klasifikasi dengan kategori yang sesuai.	Galih Ivan	Berhasil	-
UAT05	<b>UAT05 ID Referensi Fungsional:</b> F005	Irgi Haslan Putra	Berhasil	-
	<b>Deskripsi Pengujian:</b> Pengujian ini dilakukan untuk memastikan bahwa admin dapat menambahkan prompt baru untuk klasifikasi dengan benar. Tujuan pengujian adalah untuk memverifikasi bahwa sistem menerima dan memproses prompt baru tanpa error dan melakukan klasifikasi dengan benar.			

	<p><b>Skenario Pengujian:</b></p> <ol style="list-style-type: none"> <li>1. Buka browser</li> <li>2. Navigasi ke halaman Input a Prompt</li> <li>3. Masukkan teks prompt baru pada input box</li> <li>4. Klik tombol "Classify Prompt"</li> <li>5. Tunggu hingga proses klasifikasi selesai</li> </ol> <p><b>Hasil yang diharapkan:</b> Sistem menampilkan hasil klasifikasi prompt baru dengan kategori yang sesuai.</p>			
UAT06	<b>ID Referensi Fungsional:</b> F008	David Patrick Ken	Berhasil	-
	<p><b>Deskripsi Pengujian:</b> Pengujian ini dilakukan untuk memastikan bahwa admin dapat mengubah konfigurasi dan pengaturan model dengan benar. Tujuan pengujian adalah untuk memverifikasi bahwa perubahan konfigurasi disimpan dan diterapkan tanpa error.</p> <p><b>Skenario Pengujian:</b></p> <ol style="list-style-type: none"> <li>1. Buka browser</li> <li>2. Navigasi ke halaman Settings</li> <li>3. Ubah salah satu pengaturan model (misalnya, parameter learning rate)</li> <li>4. Klik tombol "Save Changes"</li> <li>5. Refresh halaman dan verifikasi bahwa perubahan disimpan</li> </ol> <p><b>Hasil yang diharapkan:</b> Sistem menyimpan dan menerapkan perubahan konfigurasi dengan sukses.</p>			



## LAMPIRAN C: HASIL PENGUJIAN LOAD TESTING

<b>ID</b>	: LOT001	
<b>Deskripsi Pengujian</b>	: Pengujian ini dilakukan untuk mengevaluasi kinerja sistem saat diakses oleh 50 pengguna secara bersamaan. Tujuannya adalah untuk memastikan bahwa sistem tetap responsif dan berfungsi dengan baik di bawah beban kerja sedang.	
<b>Jenis Metode</b>	: GET	
<b>Skenario Uji</b>	Total Users	: 50
	Ramp Up	: 50 per second
	Time	: 5 Min
<b>Response Time (ms)</b>	Min	: 1 Min(ms)
	Average	: 11.85
	Max	: 3130
<b>Requests</b>	Total Requests	: 20 per second
	Failed Requests	: 8
<b>Error rate (%)</b>	:	
<b>Chart Pengujian</b>	<p><i>Screenshot hasil Locust</i></p> 	

<b>ID</b>	: LOT002	
<b>Deskripsi Pengujian</b>	: Pengujian ini dilakukan untuk mengevaluasi kinerja sistem saat diakses oleh 5 pengguna secara bersamaan. Tujuannya adalah untuk memastikan bahwa sistem tetap responsif dan berfungsi dengan baik di bawah beban kerja rendah.	
<b>Jenis Metode</b>	: POST	
<b>Skenario Uji</b>	Total Users	: 5
	Ramp Up	: 5
	Time	: 6 Min
<b>Response Time (ms)</b>	Min	: 2 Min(ms)
	Average	: 3.49
	Max	: 42
<b>Requests</b>	Total Requests	: 730
	Failed Requests	: 730
<b>Error rate (%)</b>	:	
<b>Chart Pengujian</b>	<p><i>Screenshot hasil Locust</i></p> 