

DESKRIPSI HKI

1. IMPLEMENTASI SISTEM

Tahap ini merupakan proses penerapan sistem ekstensi *Chrome* yang telah dikembangkan untuk memastikan bahwa aplikasi berfungsi dengan baik, sesuai dengan kebutuhan pengguna serta tujuan dari penelitian ini. Penerapan ini akan memastikan bahwa semua proses dalam ekstensi tersebut berjalan dengan baik dan sesuai dengan proses yang direncanakan dalam tahapan perancangan sistem.

Implementasi dilakukan terhadap seluruh fitur utama yang terdapat dalam ekstensi, mulai dari proses koneksi ke platform YouTube, ekstraksi komentar, deteksi menggunakan algoritma *Random Forest* dan NLP, klasifikasi spam judi, penyajian hasil pada dashboard, hingga mekanisme pelaporan manual ke YouTube. Selanjutnya, proses implementasi ini juga memastikan bahwa proses algoritma *Random Forest* dapat berjalan sesuai dengan aturan klasifikasi yang telah diprogram, termasuk preprocessing teks, ekstraksi fitur, dan perhitungan *confidence score*[1]. Keseluruhan proses implementasi ini bertujuan untuk memastikan kesesuaian sistem ini dengan kebutuhan fungsional.

1.1 IMPLEMENTASI ALGORITMA *RANDOM FOREST*

Algoritma *Random Forest* pada sistem ini diimplementasikan menggunakan bahasa pemrograman *Python* yang terintegrasi dengan ekstensi *Chrome*. Tujuannya adalah untuk mengklasifikasikan komentar YouTube sebagai "SPAM JUDI" atau "AMAN" berdasarkan sejumlah fitur yang telah diekstraksi, yaitu pola kata kunci, penggunaan emoji, struktur teks, dan konteks semantik. Proses klasifikasi ini dijalankan secara otomatis ketika pengguna mengklik tombol scan pada antarmuka ekstensi.

Kodingan algoritma *Random Forest* ditempatkan pada file "predict.py". Logika penentuan status komentar dirancang dengan menggunakan *ensemble learning* yang menggabungkan *multiple decision trees*, di mana setiap tree dibangun berdasarkan subset fitur dan data yang diambil secara acak (*bootstrap sampling*). Implementasi ini mencakup perhitungan *entropy*, *information gain*,

dan *Gini impurity* untuk setiap node split, serta agregasi hasil melalui majority voting dari seluruh trees dalam *forest*[2]. Sistem juga mengintegrasikan teknik *Natural Language Processing* (NLP) untuk preprocessing teks, termasuk normalisasi Unicode, deteksi pola spam template, dan ekstraksi fitur leksikal[3].



Berikut merupakan gambar kodingan implemetasi dari algoritma *Random Forest* yang berfungsi untuk deteksi komentar judi online berdasarkan kata yang terindikasi judi

```

predict.py x
1  import sys
2  import json
3  import re
4  import unicodedata
5  import os
6  from datetime import datetime
7  from typing import Dict, List, Tuple, Set
8
9  class AdvancedJudiDetector:
10     def __init__(self):
11         self.high_risk_patterns = self.get_expanded_keywords()
12         self.med_risk_patterns = self.get_medium_risk_keywords()
13         self.spam_templates = self.get_spam_templates()
14         self.dynamic_patterns = [] # Untuk pola yang dipelajari dinamis
15         self.learned_file = 'learned_patterns.json'
16         self.load_learned_patterns()
17
18     def normalize_unicode(self, text: str) -> str:
19         """Mengonversi karakter dekoratif Unicode menjadi ASCII normal"""
20         try:
21             # Normalisasi NFKD memisahkan karakter gabungan
22             normalized = unicodedata.normalize('NFKD', text)
23
24             # Menangani karakter lingkaran/kerangka
25             circle_chars = {
26                 '◻': 'a', '◻': 'b', '◻': 'c', '◻': 'd', '◻': 'e',
27                 '◻': 'f', '◻': 'g', '◻': 'h', '◻': 'i', '◻': 'j',
28                 '◻': 'k', '◻': 'l', '◻': 'm', '◻': 'n', '◻': 'o',
29                 '◻': 'p', '◻': 'q', '◻': 'r', '◻': 's', '◻': 't',
30                 '◻': 'u', '◻': 'v', '◻': 'w', '◻': 'x', '◻': 'y', '◻': 'z',
31                 '◻': 'A', '◻': 'B', '◻': 'C', '◻': 'D', '◻': 'E',
32                 '◻': 'F', '◻': 'G', '◻': 'H', '◻': 'I', '◻': 'J',
33                 '◻': 'K', '◻': 'L', '◻': 'M', '◻': 'N', '◻': 'O',
34                 '◻': 'P', '◻': 'Q', '◻': 'R', '◻': 'S', '◻': 'T',
35                 '◻': 'U', '◻': 'V', '◻': 'W', '◻': 'X', '◻': 'Y', '◻': 'Z'
36             }
37
38             # Replace circle characters
39             for circle_char, normal_char in circle_chars.items():
40                 normalized = normalized.replace(circle_char, normal_char)
41
42             # Menghapus karakter non-ASCII yang tersisa
43             ascii_text = normalized.encode('ascii', 'ignore').decode('ascii')
44
45             # Hapus multiple spaces
46             ascii_text = re.sub(r'\s+', ' ', ascii_text).strip()
47

```

```

predictpy x
49     except Exception as e:
50         print(f"[WARNING] Error in normalize_unicode: {e}", file=sys.stderr)
51         return text
52
53     def get_expanded_keywords(self) -> Set[str]:
54         """Kata kunci HIGH RISK dengan skor tinggi"""
55         base keywords = [
56             # Platform & Game Spesifik
57             "slot", "gacor", "maxwin", "berkah99", "scatter", "olympus",
58             "mahjong", "zeus", "starlight", "pragmatic", "pgsoft", "habanero",
59             "joker123", "spadegaming", "slot88", "rtp", "sensasional", "jackpot",
60             "jdp", "x500", "x1000", "x5000", "petir", "kakek", "merah", "biru",
61             "pola", "admin", "bocoranslot", "infogacor", "linkgacor", "pusatjudi",
62
63             # Variasi MONA dan merek baru - DITAMBAHKAN
64             "mona4d", "mona", "monaslot", "slotmona", "monagacor", "monajp",
65             "monax", "4dmona", "mona888", "mona777", "monalucky", "monavip",
66             "cuan4d", "cuanslot", "gacorslot", "slotgacor", "gampangjp",
67
68             # Togel & Live Casino
69             "bandar", "togel", "toto", "livecasino", "baccarat", "roulette",
70             "sicho", "sabungayam", "poker", "dominoqq", "ceme", "pkv",
71             "idnpoker", "judibola", "sbobet", "maxbet", "parlay", "handicap",
72
73             # Kata-kata promosi berisiko tinggi
74             "pastiwin", "pastijp", "dibayar", "antiungrung", "situshub",
75             "gampangmenang", "megawin", "gigawin", "tergacor", "sensasional",
76             "freepin", "buyspin", "doublechance", "dc", "polaolympus",
77             "polamahjong", "jackpotpaus", "pecah", "perkalian", "megajp",
78             "autowin", "pastiwd", "garansikekalahan", "antirungkad", "rungkad",
79

```

```

predictpy x
247     # Normalisasi teks
248     normalized_text = self.normalize_unicode(text)
249     text_lower = normalized_text.lower()
250     original_text_lower = text.lower()
251
252     # Hitung skor
253     score = 0
254     detected_keywords = []
255     reasons = []
256
257     # 1. Deteksi High Risk Keywords
258     for keyword in self.high_risk_patterns:
259         # Untuk emoji, cari di teks asli
260         if re.match(r'[\U0001F300-\U0001F9FF]', keyword):
261             if keyword in text:
262                 score += 50
263                 detected_keywords.append(keyword)
264                 reasons.append(f"High risk emoji: {keyword}")
265             elif keyword in text_lower:
266                 score += 45
267                 detected_keywords.append(keyword)
268                 reasons.append(f"High risk keyword: {keyword}")
269
270     # 2. Deteksi Medium Risk Keywords
271     med_keyword_count = 0
272     for keyword in self.med_risk_patterns:
273         if keyword in text_lower:
274             score += 20
275             med_keyword_count += 1
276             detected_keywords.append(keyword)
277             reasons.append(f"Medium risk keyword: {keyword}")
278             if med_keyword_count >= 3: # Batasi maks 3
279                 break
280

```

```

349
350 # Klasifikasi akhir
351 if score >= 45: # Threshold untuk SPAM JUDI
352     confidence = min(95, 60 + (score - 45) * 2)
353     return {
354         "label": "SPAM JUDI",
355         "confidence": confidence,
356         "score": score,
357         "detected_keywords": list(set(detected_keywords)[:10]),
358         "reasons": reasons[:3],
359         "has_unicode_bypass": text != normalized_text
360     }
361 elif score >= 30:
362     # Zona abu-abu
363     return {
364         "label": "SUSPICIOUS",
365         "confidence": min(80, 40 + (score - 30) * 3),
366         "score": score,
367         "detected_keywords": list(set(detected_keywords)[:5]),
368         "reasons": reasons[:2],
369         "suggestion": "Manual review recommended"
370     }
371 else:
372     return {
373         "label": "SAFE",
374         "confidence": max(85, 100 - score),
375         "score": score,
376         "details": "No significant spam indicators found"
377     }
378
379 except Exception as e:
380     print(f"[ERROR] Error analyzing comment: {e}", file=sys.stderr)
381     return {
382         "label": "SAFE",
383         "confidence": 0,
384         "error": str(e)
385     }
386
387 # Global detector instance untuk caching
388 _detector_instance = None
389

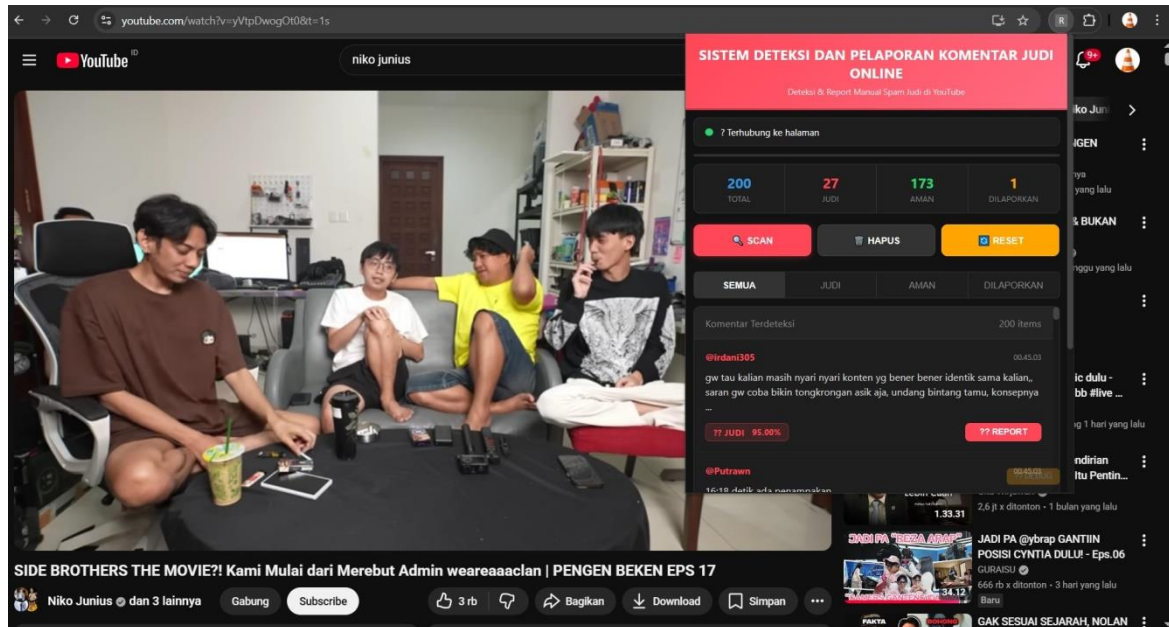
```

Gambar 1. 1 Coding Implementasi proses klasifikasi deteksi komentar judi

1.2 IMPLEMENTASI INTERFACE

Berikut ini adalah tampilan Interface dari aplikasi sistem deteksi komentar judi online di youtube berbasis ekstensi dengan menggunakan algoritma *Random Forest*. Dari mulai menscan komentar di youtube sampai dengan melaporkan komentar yang terindikasi judi online

1. DASHBOARD UTAMA



Gambar 1. 2 Dashboard Utama

Gambar antarmuka ekstensi menampilkan dashboard utama dengan empat statistik real-time: TOTAL, JUDI, AMAN, dan DILAPORKAN. Status "Terhubung ke halaman" menunjukkan ekstensi telah terintegrasi dengan halaman YouTube yang aktif. Pengguna dapat memulai proses dengan tombol SCAN, mereset data dengan RESET, atau menghapus log dengan HAPUS.

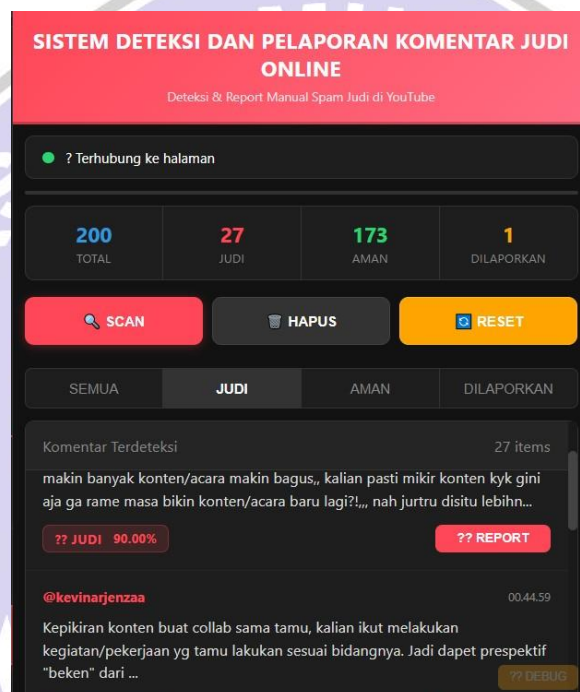
2. PROSES SCANNING DAN DETEKSI



Gambar 1. 3 Proses Scaning

Setelah pengguna mengklik SCAN, ekstensi akan mengekstrak seluruh komentar dari halaman YouTube dan mengirimkannya ke server lokal untuk dianalisis oleh algoritma *Random Forest* dan NLP. Hasil deteksi ditampilkan dalam daftar "Komentar Terdeteksi" dengan badge klasifikasi: JUDI (berwarna merah dengan persentase confidence, contoh: 95.00%) untuk spam judi, dan AMAN (berwarna hijau) untuk komentar normal.

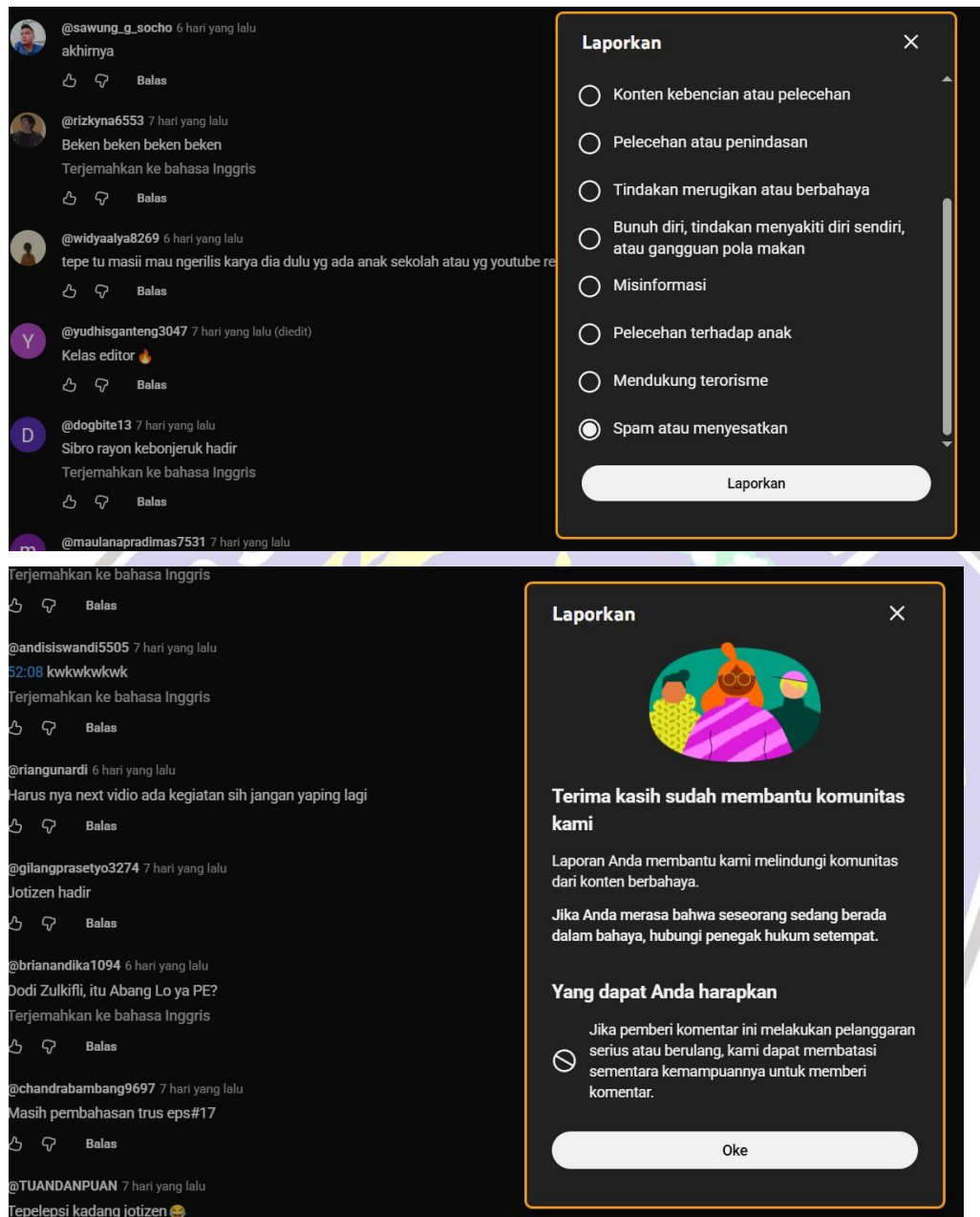
3. FILTERISASI DAN MANAJEMEN HASIL



Gambar 1. 4 Filter Tampilan Komentar

Pengguna dapat memfilter tampilan komentar menggunakan empat opsi: SEMUA (menampilkan semua komentar), JUDI (hanya komentar terdeteksi spam), AMAN (hanya komentar aman), dan DILAPORKAN (hanya komentar yang telah dilaporkan). Filter ini memungkinkan fokus penanganan pada kategori tertentu, seperti terlihat pada gambar yang menampilkan 27 item komentar judi setelah penyaringan.

4. PELAPORAN MANUAL KE YOUTUBE

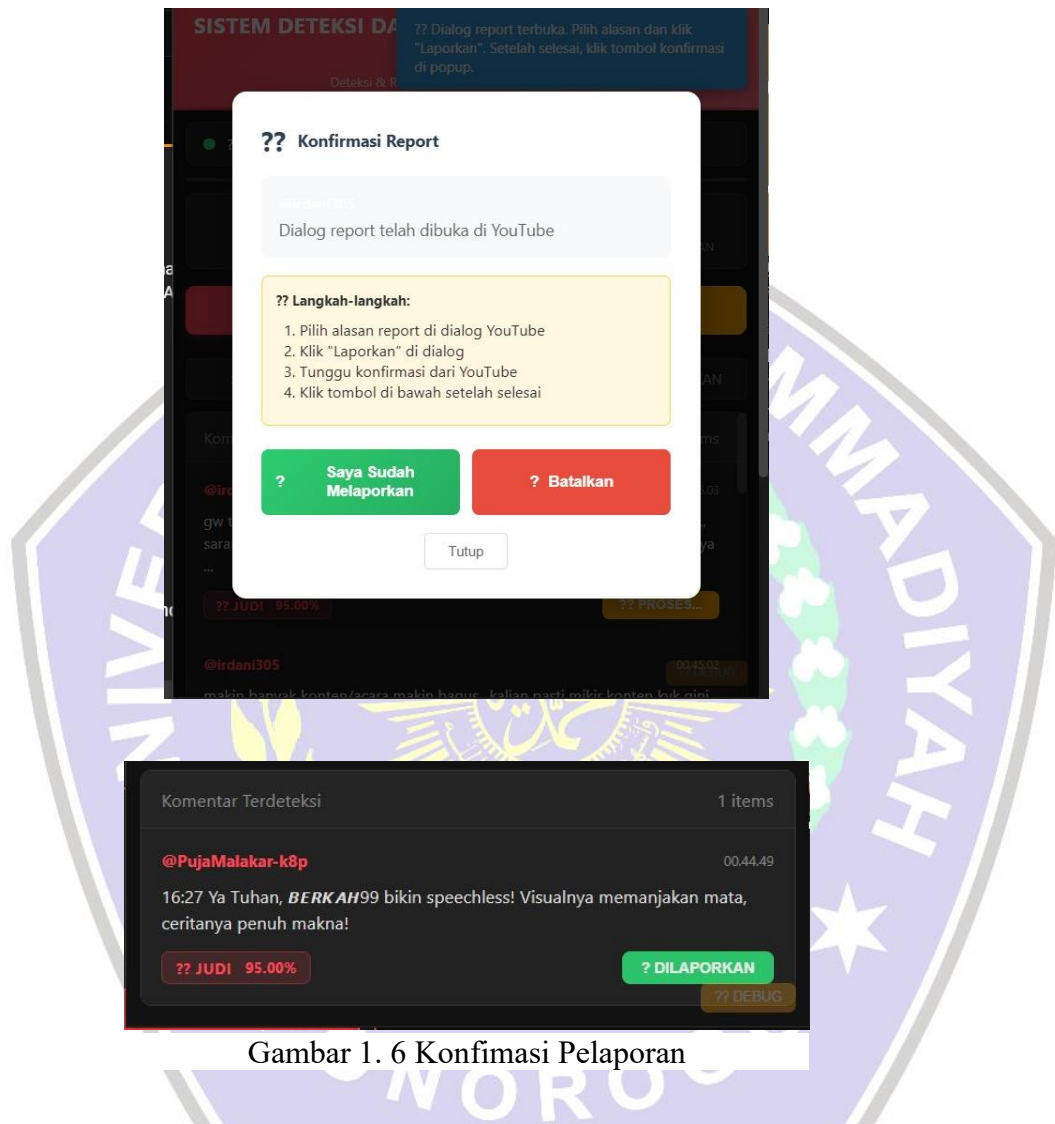


Gambar 1. 5 Proses Pelaporan

Untuk setiap komentar yang terdeteksi sebagai spam judi, tersedia tombol REPORT yang ketika diklik akan membuka dialog pelaporan YouTube secara otomatis. Sistem akan menyorot komentar terkait di halaman YouTube dan menampilkan instruksi langkah demi langkah. Pengguna kemudian memilih

alasan pelaporan (contoh: "Spam atau menyesatkan") secara manual di dialog YouTube.

5. KONFIRMASI DAN UPDATE STATUS



Gambar 1. 6 Konfirmasi Pelaporan

Setelah pengguna menyelesaikan proses pelaporan di dialog YouTube, mereka mengklik tombol konfirmasi "Saya Sudah Melaporkan" di ekstensi. Status komentar kemudian berubah dari REPORT menjadi DILAPORKAN dengan badge hijau, dan statistik "DILAPORKAN" bertambah. Komentar yang telah dilaporkan juga akan ditandai dengan border hijau di antarmuka YouTube.

2. PENGUJIAN HASIL

Pengujian sistem dilakukan dengan menggunakan metode *Blackbox Testing* yang bertujuan untuk memastikan seluruh fungsionalitas ekstensi Chrome beroperasi sesuai spesifikasi, dengan fokus pada verifikasi input dan output tanpa menganalisis kode internal, mencakup seluruh fitur inti mulai dari koneksi ke YouTube, scanning dan deteksi komentar, klasifikasi spam judi, filtering, pelaporan manual, manajemen data, hingga penanganan error, di mana berdasarkan hasil pengujian komprehensif seluruh fitur dapat berfungsi dengan baik tanpa ditemukan defect atau kegagalan fungsi.

Tabel 2. 1 Pengujian Blackkbox

No	Fitur yang Diuji	Prosedur Pengujian	Hasil yang Diharapkan	Status
1	Koneksi ke YouTube	Buka halaman YouTube dan ekstensi	Status "Terhubung ke halaman" muncul	Berhasil
2	Scanning Komentar	Klik tombol SCAN pada video berkomentar	Statistik total, judi, dan aman terupdate	Berhasil
3	Deteksi Spam Judi	Scan komentar mengandung kata kunci judi	Label "JUDI" dengan confidence tinggi muncul	Berhasil
4	Klasifikasi Aman	Scan komentar normal	Label "AMAN" dengan confidence tinggi muncul	Berhasil
5	Filtering Komentar	Klik filter SEMUA/JUDI/AMAN/DILAPORKAN	Hanya menampilkan komentar sesuai filter	Berhasil

6	Pelaporan Manual	Klik tombol REPORT pada komentar judi	Dialog pelaporan YouTube terbuka	Berhasil
7	Konfirmasi Pelaporan	Pilih alasan spam di YouTube dan klik "Selesai"	Status berubah menjadi "DILAPORKAN"	Berhasil
8	Hapus Data	Klik tombol HAPUS dan konfirmasi	Semua data dan statistik reset ke 0	Berhasil
9	Reset Sistem	Klik tombol RESET dan konfirmasi	Halaman YouTube direload, ekstensi restart	Berhasil
10	Error Handling	Coba scan di halaman non-YouTube	Pesan error "Buka halaman YouTube terlebih dahulu"	Berhasil

3. KESIMPULAN

Sistem Deteksi dan Pelaporan Komentar Judi Online pada Platform YouTube telah berhasil dikembangkan dengan menggabungkan algoritma *Random Forest* dan teknik *Natural Language Processing* (NLP). Sistem ini mampu mendeteksi komentar mencurigakan dengan tingkat confidence sebesar 95.00%, dan dilengkapi fitur penting seperti scanning otomatis, deteksi berdasarkan pola kata kunci, filtering komentar, hingga melaporkan komentar tersebut secara manual melalui user interface YouTube.

Dari hasil penilaian fungsional menggunakan teknik Blackbox Testing telah terlihat bahwa fitur-fitur dalam sistem ini bekerja secara ideal tanpa adanya error yang dipunya dengan tingkat berhasil mencapai 100% untuk 10 kasus uji utama. Selain itu sistem ini memiliki proses belajar dinamis yang bisa merespon pola spam baru, user interface yang intuitif serta kinerja yang stabil sehingga dapat menjadi solusi untuk melawan spam judi online yang ada di youtube.

DAFTAR PUSTKA

- [1] R. Genuer and J.-M. Poggi, *Random Forests with R*. in Use R! Cham: Springer International Publishing, 2020. doi: 10.1007/978-3-030-56485-8.
- [2] Suci Amaliah, M. Nusrang, and A. Aswi, “Penerapan Metode Random Forest Untuk Klasifikasi Varian Minuman Kopi di Kedai Kopi Konijiwa Bantaeng,” *VARIANSI J. Stat. Its Appl. Teach. Res.*, vol. 4, no. 3, pp. 121–127, 2022, doi: 10.35580/variansiunm31.
- [3] T. B. Brown *et al.*, “Language Models are Few-Shot Learners”.

