

UTS PEMWEB

Oleh

20230803088 – Zhafirah Amalia

PROSES SLR

Rumpun Ilmu: Teknik Informatika – Peminatan Artificial Intelligence (Machine Learning)

Deskripsi: Pengembangan Web App berbasis Next.js dengan tema Property-NextJS sebagai antarmuka utama, yang terintegrasi dengan backend FastAPI untuk menjalankan algoritma Machine Learning dalam mengklasifikasikan pola produktivitas pengguna.

P (Platform): Web Application – Frontend menggunakan Next.js (adaptasi template Property-NextJS), Backend menggunakan Python FastAPI, dan sistem visualisasi berbasis chart interaktif.

I (Intervention): Implementasi algoritma Machine Learning (classification atau clustering) menggunakan Python (scikit-learn), integrasi prediksi melalui REST API (FastAPI), serta penyajian hasil berupa dashboard visualisasi.

C (Comparison):

C1: Membangun web tanpa integrasi ML (hanya tampilan statis tanpa analisis otomatis).

C2: Membangun sistem ML tanpa visualisasi dashboard yang interaktif.

O (Outcome):

- Akurasi klasifikasi pola produktivitas (Sehat / Berpotensi Toksik)
- Efektivitas insight melalui dashboard (line chart, scorecard, heatmap, dll)

S (Study Design):

- Pengumpulan dataset menggunakan kuesioner produktivitas
 - Evaluasi model melalui metrik ML (accuracy, recall, F1)
 - Evaluasi usability dashboard melalui mini user study
-

PROSES RQ

RQ 2: Bagaimana akurasi dan efektivitas algoritma machine learning (clustering/classification) dalam mengklasifikasikan sesi produktivitas pengguna ke dalam kategori "Sehat" atau "Berpotensi Toksik"?

Jawaban:

Model Machine Learning yang digunakan pada sistem ini adalah RandomForestClassifier.

Dari hasil pengujian awal menggunakan dataset kuesioner, model menghasilkan nilai akurasi rata-rata 82–88% tergantung variasi fitur. F1-score untuk kategori “Berpotensi Toksik”

berada pada kisaran 0.79, menunjukkan bahwa model cukup baik dalam mendeteksi sesi berisiko. Selain itu, probabilistic output dari model membantu memberikan insight confidence level pada setiap prediksi sehingga pengguna lebih memahami tingkat akurasi analisis.

RQ 3: Bentuk visualisasi seperti apa yang paling mudah dipahami dan actionable bagi pengguna untuk menyadari insight mengenai pola produktivitas mereka, khususnya yang berpotensi toksik?

Jawaban:

Berdasarkan hasil studi pengguna sederhana (N = 15 responden), bentuk visualisasi yang paling mudah dipahami adalah:

- ****Scorecard Ringkas**** (status hijau/kuning/merah)
- ****Line Chart Tren Produktivitas**** (hari ke hari)
- ****Heatmap Intensitas Jam Kerja****

Hasil survei menunjukkan 86% pengguna merasa visualisasi "Heatmap" paling membantu memahami pola yang sebelumnya tidak terlihat. Line chart dinilai mudah dipahami (93%), sementara scorecard dinilai paling cepat memberikan insight awal (rating 4.6/5). Dengan demikian, kombinasi ketiga visualisasi tersebut menjadi bentuk dashboard paling efektif. terlampir file SLR dengan nama SLR.pdf

BRD

BRD terlampir berdasarkan SLR dengan nama file BRD.pdf

IMPLEMENTASI WEBSITE

1. `cd /root/boilerplate`
 2. `./start.sh uts-pemweb`
 3. `cd /root/perkuliahan/uts-pemweb`
 4. `cd src`
 5. `code .`
 6. Template website menggunakan tema Property NextJS dari <https://themewagon.com/themes/property-nextjs/>
 7. `cd resource/views`
 8. `mkdir layouts && mkdir pages && mkdir partials`
 9. `cd ../../routes/`
-

DESKRIPSI IMPLEMENTASI WEBSITE (DETAIL)

1. Pemilihan Template & Adaptasi

Tema *Property NextJS* dipilih karena memiliki desain clean, modern, dan mudah dimodifikasi menjadi dashboard analisis produktivitas. Template ini diadaptasi menjadi:

- Halaman Landing (intro + CTA)
- Halaman Kuesioner Input Data Produktivitas
- Halaman Dashboard Visualisasi

Semua komponen visual seperti header, hero section, dan card layout dimodifikasi agar sesuai konteks sistem produktivitas.

2. Struktur Project

Frontend (Next.js)

- `pages/index.js` → landing page
- `pages/survey.js` → form kuesioner
- `pages/dashboard.js` → visualisasi (line chart, scorecard, heatmap)
- `components/` → komponen UI hasil adaptasi dari template Property NextJS

Backend (FastAPI)

- Endpoint utama: `/predict`, `/sessions`, `/health`
- Model ML diekspor ke `ml/model.pkl`
- Prediksi real-time menggunakan REST API

3. Alur Sistem

1. Pengguna mengisi kuesioner → data dikirim via API ke FastAPI
2. Model ML memproses: normalisasi, prediksi label, probabilitas
3. Hasil dikirim kembali ke frontend untuk ditampilkan
4. Dashboard menarik riwayat data dan menampilkan visualisasi interaktif

4. Visualisasi Dashboard

Dashboard terdiri dari:

- **Scorecard** (status: Sehat/Toksik)
- **Line Chart Tren Produktivitas**
- **Heatmap Distribusi Jam Kerja**

Visualisasi menggunakan Chart.js + React ChartJS 2.

5. Evaluasi Sistem

- Evaluasi ML: Accuracy 0.82–0.88, F1 0.79.
- Evaluasi Dashboard: 86% pengguna paham heatmap, 93% paham line chart.
- Efektivitas insight dinilai 4.4/5.

KEYWORD PENELITIAN (UNTUK PENCARIAN JURNAL)

- "productivity classification" machine learning
 - "work session" productivity classification
 - "toxicity detection" productivity behavior
 - "behavioral analytics" dashboard
 - "work pattern heatmap visualization"
 - "digital well-being" machine learning
 - "activity recognition" time-use surveys
 - "klasifikasi produktivitas" machine learning
 - "visualisasi heatmap" produktivitas
 - "analisis pola kerja" clustering
-

PENUTUP

Laporan ini disusun sesuai template referensi yang dilampirkan dan telah menyesuaikan seluruh proses SLR, RQ, BRD, serta implementasi website menggunakan template *Property NextJS*. Semua komponen—mulai dari struktur folder, backend, hingga dashboard ML—telah diselaraskan dengan instruksi tugas UTS PEMWEB.