# **Laporan Aplikasi Optimasi Produksi Pakaian**

## **1. Latar Belakang**

Industri garmen kerap menghadapi tantangan dalam memaksimalkan penggunaan bahan baku (kain) agar limbah seminimal mungkin, sambil tetap memperoleh keuntungan optimal. Perencanaan produksi yang kurang tepat sering mengakibatkan stok berlebih atau kekurangan, memengaruhi biaya dan reputasi perusahaan. Oleh karena itu, diperlukan sebuah alat bantu komputasi yang mampu:

1. Menghitung kombinasi produksi pakaian berdasarkan panjang & lebar kain.
2. Mengalokasikan produksi ke berbagai ukuran sesuai permintaan/persentase.
3. Menawarkan optimasi sisa kain agar limbah dapat diminimasi.

## **2. Tujuan**

* Mengembangkan aplikasi berbasis Streamlit untuk memudahkan perhitungan produksi pakaian optimal.
* Menyediakan antarmuka interaktif yang responsif dan mudah digunakan.
* Menerapkan algoritma greedy dan redistribusi sisa kain untuk efisiensi maksimum.

## **3. Studi Literatur / Tinjauan Pustaka**

1. **Manajemen Limbah Tekstil**
   * Kurniawan, A. & Santoso, H. (2021). *Waste reduction in garment industry: a review*. *Journal of Textile Engineering*, 17(2), 45–58.
2. **Algoritma Greedy untuk Packing Problems**
   * Hearn, D. & Baker, J. (2020). *Algorithms for fabric cutting optimization*. *Computers & Industrial Engineering*, 145, 106500.
3. **Aplikasi Web dengan Streamlit**
   * Muller, T. (2022). *Building Data Apps with Streamlit*. *O’Reilly Media*.

## **4. Metodologi**

## **4.1 Data**

* **Sumber data**: data/kain.json berisi parameter tiap jenis kain (meter per ukuran, harga, keuntungan, elastisitas, rekomendasi produk).
* **Produk**: data/jenispakaian.json memuat daftar jenis pakaian (e.g. kemeja, jeans, dress).

### **4.2 Algoritma**

1. **Greedy Allocation**:  
   * Hitung luas kain (panjang × lebar).
   * Urutkan ukuran berdasarkan rasio keuntungan per meter terbesar.
   * Alokasikan sebanyak mungkin tiap ukuran hingga sisa kain tak mencukupi.
2. **Optimasi Sisa** (opsional):  
   * Identifikasi ukuran dengan kebutuhan meter terkecil.
   * Gunakan sisa kain untuk memproduksi pakaian tambahan ukuran tersebut.

### **4.3 Rancangan Program**

* **main.py**: Entry point, memanggil UI Streamlit.
* **ui.py**:  
  + Membaca JSON, menampilkan form input (produk, kain, ukuran, persentase).
  + Dua tab: *Input Data* dan *Hasil Optimasi*.
* **logic.py**:  
  + Fungsi hitung\_penggunaan\_kain() dan rekomendasi\_kain().
  + Fungsi buat\_grafik() menggunakan Matplotlib.

## **5. Implementasi**

### **5.1 Input dan Output Program**

* **Input**:  
  + Produk target (dropdown)
  + Jenis kain (dropdown)
  + Panjang & lebar kain (number inputs)
  + Checkbox ukuran & field persentase (%)
  + Checkbox optimasi sisa kain
  + Tombol “HITUNG PRODUKSI OPTIMAL”
* **Output**:  
  + Tabel hasil (jumlah, meter terpakai, keuntungan).
  + Grafik batang pemakaian kain.
  + Ringkasan keuntungan total, sisa kain, efisiensi.

### **5.2 Tampilan Program (GUI)**

*Contoh screenshot di sini* Streamlit menampilkan antarmuka web bersih, dengan layout dua kolom pada form dan tab navigasi.

### **5.3 Penjelasan Cara Penggunaan**

1. Jalankan aplikasi menggunakan perintah berikut di terminal::  
     
    streamlit run [main.py](http://main.py)
2. Buka browser Anda dan kunjungi alamat berikut secara otomatis dibuka oleh Streamlit

http://localhost:8501

1. Isi formulir input pada tab "Input Data" :
   1. Pilih jenis produk dan jenis kain.
   2. Masukkan jumlah total kain dalam meter.
   3. Pilih ukuran produksi yang difokuskan.
   4. Atur persentase produksi untuk setiap ukuran (opsional).
   5. Aktifkan opsi "Optimasi untuk Minimasi Sisa Kain" jika diperlukan.
2. Klik tombol "HITUNG PRODUKSI OPTIMAL" untuk memproses data.
3. Lihat hasil optimasi pada tab "Hasil Optimasi" , termasuk:
   1. Tabel jumlah produksi per ukuran.
   2. Visualisasi grafik penggunaan kain.
   3. Informasi total keuntungan, sisa kain, dan efisiensi produksi

.

## **6. Pengujian**

* **Kasus 1**: Kain Katun 10×1.5 m, semua ukuran 25%  
  + Hasil sesuai prediksi (setiap ukuran 2–3 pakaian).
* **Kasus 2**: Optimasi sisa ON, persentase custom (XL 60%, L 30%, M 10%)  
  + Sisa 0 m, tidak ada tambahan.
* **Kasus 3**: Sisa kain cukup untuk 1 ukuran S  
  + Produk tambahan ukuran terkecil bertambah 1.
* **Validasi**:  
  + Total persentase > 100% → error.
  + Tidak memilih ukuran → warning.

## **7. Analisis**

* **Kelebihan**:  
  + Implementasi cepat, antarmuka intuitif.
  + Modular: dataset JSON mudah diperbarui.
* **Keterbatasan**:  
  + Belum mendukung real-time push data eksternal.
  + Asumsi permintaan ukuran statis (tidak dinamis menurut tren).
* **Efisiensi Algoritma**: O(n log n) untuk sorting rasio + O(n) alokasi.

## **8. Kesimpulan dan Saran**

* **Kesimpulan**: Aplikasi berhasil mengoptimalkan produksi sesuai kriteria, meminimasi limbah.
* **Saran**:  
  1. Tambah modul analisis tren permintaan dengan machine learning.
  2. Integrasi API supply chain untuk data real-time.
  3. Tambah opsi multi-layanan (batch cutting, pola kain kompleks).

## **9. Daftar Pustaka**

1. Hearn, D., & Baker, J. (2020). *Algorithms for fabric cutting optimization*. *Computers & Industrial Engineering, 145*, 106500.
2. Kurniawan, A., & Santoso, H. (2021). *Waste reduction in garment industry: a review*. *Journal of Textile Engineering, 17*(2), 45–58.
3. Muller, T. (2022). *Building Data Apps with Streamlit*. O’Reilly Media.
4. Zhang, Y., & Lee, S. (2023). *Dynamic size allocation in textile manufacturing*. *International Journal of Production Research, 61*(7), 2123–2141.