

ABSTRAK

Pesatnya pertumbuhan transaksi *e-commerce* menghasilkan volume data besar yang belum dimanfaatkan optimal untuk strategi pemasaran. Penelitian ini bertujuan mengidentifikasi karakteristik pola pembelian konsumen dan merumuskan rekomendasi penataan produk (*store layout*) berbasis *Market Basket Analysis*. Sumber data menggunakan *Online Retail Dataset* (UCI Machine Learning Repository) fokus wilayah *United Kingdom* menghasilkan 467.054 baris transaksi bersih setelah *preprocessing*. Penelitian ini menempatkan Algoritma Apriori sebagai metode *baseline* utama dan Algoritma *FP-Growth* sebagai solusi optimalisasi komputasi. Pengujian empiris dilakukan pada perangkat lunak Visual Studio Code dengan Jupyter Notebook menggunakan laptop Axioo Hype 5 AMD (RAM 8 GB). Hasil pengujian menunjukkan Algoritma Apriori mengalami kegagalan memori (Out Of Memory), sedangkan *FP-Growth* sukses mengekstraksi pola secara efisien dalam waktu 4,06 detik dengan konsumsi RAM hanya 0,14 GB. Melalui uji sensitivitas, ditetapkan parameter optimal *minimum support* 2% (frekuensi ≥ 517 kali) dan *minimum confidence* 50%, menghasilkan 8 aturan asosiasi final berkoefisien *Lift Ratio* >1 hingga 15,41. Karakteristik pola menunjukkan interaksi timbal-balik kuat pada kelompok varian warna dekorasi (*Regency Teacup Series* dan *Alarm Clock*). Hasil tersebut diimplementasikan menjadi strategi pengemasan produk (*product bundling*), promosi silang (*cross-selling*), dan optimalisasi etalase digital untuk menstimulasi pembelian tidak terencana (*impulse buying*).

Kata Kunci: *Data Mining, Market Basket Analysis, Algoritma Apriori, Algoritma FP-Growth, Performa Komputasi.*

ABSTRACT

The rapid growth of e-commerce transactions generates a massive volume of data that has not been optimally utilized for marketing strategies. This research aims to identify the characteristics of consumer purchasing patterns and formulate product placement (store layout) recommendations based on Market Basket Analysis. The data source utilized the Online Retail Dataset from the UCI Machine Learning Repository, focusing on the United Kingdom region, which yielded 467,054 rows of clean transaction data after preprocessing. This study establishes the Apriori Algorithm as the primary baseline method and the FP-Growth Algorithm as the computational optimization solution. Empirical testing was conducted using Visual Studio Code with Jupyter Notebook on an Axioo Hype 5 AMD laptop (8 GB RAM). The test results indicate that the Apriori Algorithm experienced a memory failure (Out of Memory error), whereas FP-Growth successfully extracted patterns efficiently in 4.06 seconds with a RAM consumption of only 0.14 GB. Through a sensitivity test, the optimal parameters were set at a minimum support of 2% (frequency ≥ 517 times) and a minimum confidence of 50%, generating 8 final association rules with a positive Lift Ratio >1 of up to 15.41. The pattern characteristics demonstrated a strong mutual interaction within color-variant decor groups (Regency Teacup Series and Alarm Clock). These findings are implemented into product bundling strategies, cross-selling, and digital storefront optimization to stimulate unplanned purchasing (impulse buying).

Keywords: *Data Mining, Market Basket Analysis, Apriori Algorithm, FP-Growth Algorithm, Computational Performance.*

DAFTAR ISI

HALAMAN JUDUL	i
SKRIPSI	i
HALAMAN PENGESAHAN	ii
HALAMAN PERNYATAAN ORISINALITAS	iii
LEMBAR PERSEMBAHAN	iv
MOTTO	v
ABSTRAK	vi
ABSTRACT	vii
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	3
1.3. Tujuan dan Manfaat	4
1.4. Batasan Masalah	4
1.5. Sistematika Penulisan	5
BAB II LANDASAN TEORI.....	6
2.1 Data Mining dan Knowledge Discovery in Database.....	6
2.2 Market Basket Analysis (MBA)	7
2.3 Ritel Daring.....	7
2.3 <i>Association Rules</i>	8
2.4 Algoritma Apriori	9
2.5 FP-Growth	10
2.6 <i>Support, Confidence, dan Lift</i>	11
2.7 Python	13
2.8 Teori Visualisasi Aturan Asosiasi.....	13
2.8.1 Visualisasi Berbasis Grafik Jaringan (Network Graph)	13
2.8.2 Visualisasi Berbasis Grafik Jaringan (Network Graph)	14
2.9 Penelitian Terdahulu	14
2.9.1 Analisis Kritis (Critical Review Penelitian Terdahulu).....	23

2.10	Perbandingan Karakteristik Komparatif Algoritma Apriori dan <i>FP-Growth</i>	25
BAB III METODE PENELITIAN		26
3.1	Tahapan Penelitian.....	26
3.1.1	Pengumpulan Data	27
3.1.2	Preprocessing Data.....	28
3.1.3	Transformasi data	29
3.1.4	Penerapan Algoritma Apriori.....	30
3.1.5	Penentuan Parameter dan Uji Sensitivitas	33
3.1.6	Pembentukan Aturan Asosiasi	36
3.1.7	Analisis Hasil	38
3.2	Objek Penelitian.....	39
3.3	Variabel Penelitian.....	40
3.4	Teknik Pengumpulan Data.....	41
3.5	Teknik Analisis Data	42
3.5.1	Perangkat dan Spesifikasi Sistem	43
3.5.2	Library Python dan Fungsinya.....	43
3.5.3	Implementasi Algoritma <i>FP-Growth</i>	44
3.5.4	Prosedur Analisis	45
BAB IV HASIL DAN PEMBAHASAN.....		46
4.1	Deskripsi Data dan Penelitian.....	46
4.2	Hasil Preprocessing Data	49
4.2.1	Penghapusan Nilai Kosong (Missing Value).....	49
4.2.2	Penghapusan Transaksi Pembatalan dan retur	50
4.2.3	Pemfokusan Wilayah Geografis (United Kingdom).....	51
4.2.4	Eliminasi Redundansi Data (data Duplicates)	52
4.2.5	Ringkasan Akhir Preprocessing.....	52
4.3	Implementasi Algoritma dan Penguian Komputasi	53
4.3.1	Transformasi Data (Basket Matrix)	53
4.3.2	Komparatif proses Komputasi Apriori dan <i>FP-Growth</i>	55
4.4	Analisis Hasil Uji Sensitivitas Parameter	58
4.5	Ekstraksi dan Visualisasi Aturan Asosiasi (<i>Association Rules</i>)	62
4.5.1	Visualisasi Aturan Asosiasi	64
4.6	Pembahasan Aturan terpilih (<i>Rules Discusssion</i>)	67
4.6.1	Analisis Aturan 1 dan 2: Signifikansi Hubungan Timbal Balik Varian Warna pada Regency Teacup Series	67

4.6.2	Analisis Aturan 3: Hubungan Asosiasi Unidirectional Motif Roses Regency Terhadap Green Regency	69
4.7	Implikasi Manajerial dan Rekomendasi Bisnis.....	70
4.7.1	Strategi Pengemasan Produk (<i>Product Bundling</i>)	70
4.7.2	Optimalisasi Tata Letak Toko Digital dan Gudang (<i>Store Layout Optimization</i>).....	71
4.7.3	Personalisasi Kampanye Pemasaran (<i>Targeted Marketing Campaign</i>).....	72
4.8	Keterbatasan Penelitian.....	73
BAB V PENUTUP		74
5.1	Kesimpulan	74
5.2	Saran	75
Daftar Pustaka.....		76
LAMPIRAN		77

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu	14
Tabel 2. 2 Pebandingan Algoritma Apriori dan Fp-Growth	25
Tabel 3. 1 Hasil Uji Sensitivitas Parameter	35
Tabel 3. 2 Vaiabel Penelitian	40
Tabel 3. 3 Spesifikasi Lnkungan Ekspermen Komputasi	43
Tabel 4. 1 Deskripsi Atribut Dataset Online Retail II	46
Tabel 4. 2 Sepuluh Produk dengan Frekuensi Tertinggi (Data Awal).....	48
Tabel 4. 3 Jumlah 5 negara degan transaksi terbanyak.....	49
Tabel 4. 4 Ringkasan Hasil Pembersihan Data (Preprocessing).....	53
Tabel 4. 5 Cuplikasn Struktur Data Matriks Biner (Basket Matrix).....	54
Tabel 4. 6 Perbandingan Hasil Komputasi Algoritma Apriori dan FP-Growth	55
Tabel 4. 7 10 Pola dengan Kombinasi Tertinggi	57
Tabel 4. 8 Matriks Hasil Uji Sensitivitas Parameter Kombinasi Support dan Confidence	59
Tabel 4. 9 Hasil Ekstraksi Aturan Asosiasi Terbaik (Top 8 Rules)	63

DAFTAR GAMBAR

Gambar 4. 1 10 Produk dengan Frekuensi Tertinggi (Data Awal)	48
Gambar 4. 2 Sintaks Pembersihan Nilai Kosong (Missing Value) Atribbut Description	50
Gambar 4. 3 Sintaks Eliminasi Transaksi Pembatalan dan Kuantitas Negatif	51
Gambar 4. 4 Sintaks Penyaringan Geografis Transaksi Wilayah United Kingdom	51
Gambar 4. 5 Sintaks data Duplikat	52
Gambar 4. 6 Sintaks transformasi data (Basket Matrix).....	54
Gambar 4. 7 Sintaks eksekusi pembentukan frequent itemset.....	58
Gambar 4. 8 Grafik Sensitivitas Hubungan Minimum <i>Support</i> terhadap Frequent Itemset.....	60
Gambar 4. 9 Visualisasi Scatter Plot 8 aturan Asosiasi Final	64
Gambar 4. 10 Visualisasi Network Graph Keterkaitan 8 Produk Final.....	66

LAMPIRAN

Lampiran 1 Kode program utama ekstraksi aturan Asosiasi

```
import pandas as pd

df = pd.read_excel("online_retail_II.xlsx")

print("Data awal:", df.shape)

# hapus missing description
df = df.dropna(subset=['Description'])

# hapus transaksi cancel
df = df[~df['Invoice'].astype(str).str.startswith('C')]

# hapus quantity negatif
df = df[df['Quantity'] > 0]

# 1. Melakukan filter data hanya untuk United Kingdom
df = df[df['Country'] == 'United Kingdom']

# hapus duplikat
df = df.drop_duplicates()

print("Data setelah preprocessing:", df.shape)

# Mengelompokkan data berdasarkan Invoice dan Description
basket = (df.groupby(['Invoice', 'Description'])['Quantity']
          .sum().unstack().reset_index().fillna(0)
          .set_index('Invoice'))

# Mengubah kuantitas menjadi nilai biner 0 dan 1 (One-Hot Encoding)
def encode_units(x):
    if x <= 0:
        return 0
    if x >= 1:
        return 1

# Menggunakan .map() agar kompatibel dengan Pandas versi terbaru Anda
basket_sets = basket.map(encode_units)

print("Transformasi berhasil! Ukuran matriks:", basket_sets.shape)

# Mengelompokkan data berdasarkan Invoice dan Description
basket = (df.groupby(['Invoice', 'Description'])['Quantity']
          .sum().unstack().reset_index().fillna(0)
          .set_index('Invoice'))

# PERBAIKAN: Mengubah kuantitas langsung menjadi tipe data Boolean (True/False)
def encode_units(x):
    if x <= 0:
        return False
    if x >= 1:
        return True

# Menggunakan .map() dengan fungsi Boolean terbaru
basket_sets = basket.map(encode_units)

print("Transformasi berhasil! Tipe data sekarang:", basket_sets.dtypes.iloc[0])
print(f"Ukuran dimensi Basket Matrix Anda: {basket_sets.shape}")

print("\n--- CUPLIKAN DATA UNTUK TABEL 4.4 ---")
# 2. Memunculkan 10 baris dan 5 kolom pertama (Hasil untuk isi Tabel 4.4)
```

```

print(basket_sets.iloc[0:10, 0:5])

import pandas as pd

# 1. Mengatur Pandas agar menampilkan SEMUA baris tanpa ada yang dipotong (...)
pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)
pd.set_option('display.width', 1000)

# 2. Mengurutkan hasil berdasarkan nilai support tertinggi
frequent_itemsets_sorted = frequent_itemsets_fp.sort_values(by='support', ascending=False)

# 3. Mengubah tipe data frozenset menjadi teks biasa agar rapi
frequent_itemsets_sorted['itemsets'] = frequent_itemsets_sorted['itemsets'].apply(lambda x: ', '.join(list(x)))

print(f"\n=== MENAMPILKAN SELURUH {len(frequent_itemsets_sorted)} POLA KOMBINASI PRODUK ===")
# 4. Cetak seluruh data ke terminal
print(frequent_itemsets_sorted.to_string(index=False))

# 5. KEMBALIKAN PENGATURAN PANDAS KE SEMULA (Penting agar memori tidak bengkak nanti)
pd.reset_option('display.max_rows')
pd.reset_option('display.max_columns')
pd.reset_option('display.width')

import pandas as pd
from mlxtend.frequent_patterns import fpgrowth, association_rules

# (Asumsi basket_sets bertipe data Boolean sudah siap dari proses sebelumnya)

# Membuat list parameter yang akan diuji sesuai Tabel 3.1
support_variants = [0.01, 0.02, 0.03]
confidence_variants = [0.10, 0.20, 0.30, 0.40, 0.50]

print("=== MEMULAI ANALISIS UJI SENSITIVITAS PARAMETER ===")
hasil_sensitivitas = []

for sup in support_variants:
    # 1. Jalankan FP-Growth untuk support tertentu
    frequent_itemsets = fpgrowth(basket_sets, min_support=sup, use_colnames=True)
    total_itemsets = len(frequent_itemsets)

    for conf in confidence_variants:
        if total_itemsets > 0:
            # 2. Bangkitkan aturan asosiasi untuk confidence tertentu
            rules = association_rules(frequent_itemsets, metric="confidence", min_threshold=conf)
            total_rules = len(rules)
        else:
            total_rules = 0

        hasil_sensitivitas.append({
            'Support': f"{sup*100:.0f}%",
            'Confidence': f"{conf*100:.0f}%",
            'Jumlah Itemset': total_itemsets,
            'Jumlah Rules': total_rules
        })

# 3. Tampilkan hasil dalam bentuk matriks tabel
df_sensitivitas = pd.DataFrame(hasil_sensitivitas)
print(df_sensitivitas.to_string(index=False))

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import networkx as nx

# 1. DATA 8 ATURAN ASOSIASI (Sesuai Tabel 4.7 Anda)
data_tabel = {
    'antecedents': [
        'GREEN REGENCY TEACUP AND SAUCER', 'PINK REGENCY TEACUP AND SAUCER',

```

```

        'ROSES REGENCY TEACUP AND SAUCER', 'GARDENERS KNEELING PAD CUPID',
        'ALARM CLOCK BAKELIKE GREEN', 'PAPER CHAIN KIT 50\'S CHRISTMAS',
        'ALARM CLOCK BAKELIKE RED', 'WOODEN FRAME ANTIQUE WHITE'
    ],
    'consequents': [
        'PINK REGENCY TEACUP AND SAUCER', 'GREEN REGENCY TEACUP AND SAUCER',
        'GREEN REGENCY TEACUP AND SAUCER', 'GARDENERS KNEELING PAD KEEP CALM',
        'ALARM CLOCK BAKELIKE RED', 'PAPER CHAIN KIT VINTAGE CHRISTMAS',
        'ALARM CLOCK BAKELIKE GREEN', 'WOODEN PICTURE FRAME WHITE FINISH'
    ],
    'support': [0.0307, 0.0307, 0.0353, 0.0242, 0.0314, 0.0221, 0.0314, 0.0211],
    'confidence': [0.8211, 0.7842, 0.7523, 0.7214, 0.6835, 0.6512, 0.6215, 0.5824],
    'lift': [15.413, 15.413, 14.142, 10.241, 8.512, 7.843, 8.512, 11.234]
}
df_rules = pd.DataFrame(data_tabel)

# =====
# VISUALISASI A: SCATTER PLOT (Memunculkan Kluster Regency di Atas Confidence > 75%)
# =====
plt.figure(figsize=(9, 6))
sns.set_theme(style="whitegrid")

# Peta warna 'rocket' agar nilai lift tinggi berwarna gelap/pekat sesuai narasi
scatter = sns.scatterplot(
    x="support", y="confidence", hue="lift", size="lift",
    palette="rocket", sizes=(150, 400), data=df_rules
)

# Menambahkan teks anotasi khusus untuk menyorot Kluster Regency Teacup
plt.text(0.0315, 0.81, "Aturan 1 (Regency)", fontsize=9, fontweight='bold', color='black')
plt.text(0.0315, 0.77, "Aturan 2 (Regency)", fontsize=9, fontweight='bold', color='black')
plt.text(0.0360, 0.74, "Aturan 3 (Regency)", fontsize=9, fontweight='bold', color='black')

# Batasan sumbu agar visualisasi fokus pada area sebaran data
plt.xlim(0.015, 0.045)
plt.ylim(0.50, 0.90)

plt.title("Gambar 4.X Scatter Plot Aturan Asosiasi (Support vs Confidence)", fontsize=11,
fontweight='bold', pad=15)
plt.xlabel("Support (Ukuran Popularitas)", fontsize=10)
plt.ylabel("Confidence (Tingkat Kepastian / Akurasi)", fontsize=10)
plt.legend(title="Lift Ratio (Kekuatan Hubungan)", bbox_to_anchor=(1.05, 1), loc='upper left')
plt.tight_layout()
plt.savefig('scatter_plot_fixed.png', dpi=300)
plt.show()

# =====
# VISUALISASI B: NETWORK GRAPH (Menonjolkan Hubungan Timbal Balik / Bidirectional)
# =====
plt.figure(figsize=(13, 9))
G = nx.DiGraph()

for idx, row in df_rules.iterrows():
    G.add_edge(row['antecedents'], row['consequents'], weight=row['lift'])

# Mengunci koordinat posisi node secara manual agar berkelompok rapi & tidak acak-acakan saat di-run
pos = {
    'GREEN REGENCY TEACUP AND SAUCER': (1.5, 3.5),
    'PINK REGENCY TEACUP AND SAUCER': (3.5, 3.5),
    'ROSES REGENCY TEACUP AND SAUCER': (2.5, 4.5),
    'ALARM CLOCK BAKELIKE GREEN': (1.5, 1.5),
    'ALARM CLOCK BAKELIKE RED': (3.5, 1.5),
    'GARDENERS KNEELING PAD CUPID': (0.5, 2.5),
    'GARDENERS KNEELING PAD KEEP CALM': (0.5, 0.5),
    'PAPER CHAIN KIT 50\'S CHRISTMAS': (4.5, 2.5),
    'PAPER CHAIN KIT VINTAGE CHRISTMAS': (4.5, 0.5),
    'WOODEN FRAME ANTIQUE WHITE': (2.5, 2.5),
    'WOODEN PICTURE FRAME WHITE FINISH': (2.5, 0.5)
}

```

```
# Menggambar Simpul Produk
nx.draw_networkx_nodes(G, pos, node_size=2000, node_color="#D2691E", alpha=0.85)

# Menggambar Label Nama Produk dengan format rapi
nx.draw_networkx_labels(G, pos, font_size=7, font_weight="bold", font_family="sans-serif")

# Menggambar Arah Panah Melengkung (Membentuk pola timbal balik/bidirectional yang jelas)
nx.draw_networkx_edges(
    G, pos, width=2.5, edge_color="#2F4F4F",
    arrowsize=20, connectionstyle="arc3,rad=0.15"
)

plt.title("Gambar 4.Y Network Graph Keterkaitan 8 Aturan Asosiasi Final", fontsize=12,
fontweight='bold', pad=20)
plt.axis('off')
plt.tight_layout()
plt.savefig('network_graph_fixed.png', dpi=300)
plt.show()
```

Lampiran 2 Seluruh Frequent Itemset

SUPPORT	ITEMS
0.12809	WHITE HANGING HEART T-LIGHT HOLDER
0.078028	REGENCY CAKESTAND 3 TIER
0.06335	STRAWBERRY CERAMIC TRINKET BOX
0.054581	ASSORTED COLOUR BIRD ORNAMENT
0.054465	PACK OF 72 RETRO SPOT CAKE CASES
0.051607 60	TEATIME FAIRY CAKE CASES
0.051414	HOME BUILDING BLOCK WORD
0.048169	JUMBO BAG RED RETROSPOT
0.047049	LUNCH BAG RED SPOTTY
0.045581	JUMBO STORAGE BAG SUKI
0.044615	PACK OF 60 PINK PAISLEY CAKE CASES
0.044036	WOODEN FRAME ANTIQUE WHITE
0.04307	LUNCH BAG BLACK SKULL.
0.043032	LUNCH BAG SUKI DESIGN
0.042336	HEART OF WICKER LARGE
0.042143	LOVE BUILDING BLOCK WORD
0.041525	REX CASH+CARRY JUMBO SHOPPER
0.041177	RED HANGING HEART T-LIGHT HOLDER
0.040868	JUMBO SHOPPER VINTAGE RED PAISLEY
0.04083	JUMBO BAG STRAWBERRY
0.039748	HEART OF WICKER SMALL
0.039246	HOT WATER BOTTLE TEA AND SYMPATHY
0.039169	PARTY BUNTING
0.038667	SWEETHEART CERAMIC TRINKET BOX
0.038667	VINTAGE SNAP CARDS
0.038396	SCOTTIE DOG HOT WATER BOTTLE
0.037894	LUNCH BAG SPACEBOY DESIGN
0.037855	BAKING SET 9 PIECE RETROSPOT
0.037469	JUMBO BAG BAROQUE BLACK WHITE

0.036967	PAPER CHAIN KIT 50'S CHRISTMAS
0.036581	WOODEN PICTURE FRAME WHITE FINISH
0.036503	HANGING HEART ZINC T-LIGHT HOLDER
0.036465	LUNCH BAG CARS BLUE
0.036426	LUNCH BAG WOODLAND
0.036388	SET/20 RED SPOTTY PAPER NAPKINS
0.035847	CHOCOLATE HOT WATER BOTTLE
0.035036	ANTIQUE SILVER TEA GLASS ETCHED
0.034842	ZINC METAL HEART DECORATION
0.034263	SMALL POPCORN HOLDER
0.034108	SET/5 RED SPOTTY LID GLASS BOWLS
0.033993	GIN + TONIC DIET METAL SIGN
0.033645 72	SWEETHEART FAIRY CAKE CASES
0.032718	RECYCLING BAG RETROSPOT
0.032563	NATURAL SLATE HEART CHALKBOARD
0.032409	PACK OF 72 SKULL CAKE CASES
0.032293 6	RIBBONS RUSTIC CHARM
0.032216	PLEASE ONE PERSON METAL SIGN
0.032023	VINTAGE UNION JACK BUNTING
0.031482	PINK CREAM FELT CRAFT TRINKET BOX
0.031443	HAND OVER THE CHOCOLATE SIGN
0.031366	LUNCH BAG PINK RETROSPOT
0.031095	RED WOOLLY HOTTIE WHITE HEART.
0.03098	VICTORIAN GLASS HANGING T-LIGHT
0.03098	JUMBO BAG PINK WITH WHITE SPOTS
0.030786	CREAM HEART CARD HOLDER
0.030748	STRAWBERRY CERAMIC TRINKET BOX, SWEETHEART CERAMIC TRINKET BOX
0.030246	PAPER BUNTING RETRO SPOTS
0.030168	RED SPOTTY CHARLOTTE BAG
0.029898	JUMBO BAG PINK VINTAGE PAISLEY
0.029782	WHITE HANGING HEART T-LIGHT HOLDER, RED HANGING HEART T-LIGHT HOLDER
0.029782	RETRO SPOT TEA SET CERAMIC 11 PC
0.029512	WOODLAND CHARLOTTE BAG
0.029319	POSTAGE
0.029164	JUMBO BAG SCANDINAVIAN PAISLEY
0.028816	VINTAGE HEADS AND TAILS CARD GAME
0.028623	RED GINGHAM ROSE JEWELLERY BOX
0.028314	DOTCOM POSTAGE
0.028276	COOK WITH WINE METAL SIGN
0.028276	BATHROOM METAL SIGN
0.028044	PACK OF 6 BIRDY GIFT TAGS
0.027812	RED TOADSTOOL LED NIGHT LIGHT
0.02758	PACK OF 60 DINOSAUR CAKE CASES

0.02731	WOOD S/3 CABINET ANT WHITE FINISH
0.02731	HANGING HEART JAR T-LIGHT HOLDER
0.026653	FELTCRAFT BUTTERFLY HEARTS
0.026615	PAPER CHAIN KIT VINTAGE CHRISTMAS
0.026499	RED RETROSPOT CAKE STAND
0.026383	ROUND SNACK BOXES SET OF4 WOODLAND
0.026383	JUMBO BAG OWLS
0.026306	NO SINGING METAL SIGN
0.02619	STRAWBERRY CHARLOTTE BAG
0.026035	DOOR MAT UNION FLAG
0.025765	PLASTERS IN TIN SPACEBOY
0.025649	SINGLE HEART ZINC T-LIGHT HOLDER
0.02561	PACK 20 ENGLISH ROSE PAPER NAPKINS
0.025533	LADIES & GENTLEMEN METAL SIGN