



Analisis Survey eChannel Fintech eCommerce dan eLifestyle

Fernaldi Fauzie

18219099

Sistem dan Teknologi Informasi

Data Cleansing

MENGHAPUS RESPONDEN YANG MEMBUAT DUPLIKASI DATA, DATA YANG DISIMPAN MERUPAKAN DATA YANG TERBARU

```
responden = pd.DataFrame(df, columns=['Nama Responden'])
indeksResponden = responden[responden.duplicated()]
#display(indeksResponden) untuk mencari indeks data duplikasi
df = df.drop([20,43,61,96,97,102,103,137,150,155,156,168,169,172,188,222,223,224,243,253,256,265,274,280,298,305,312,313,324])
```

Mengolah Data Surveyor untuk Data Cleansing

DATA SURVEYOR

```
# Merapikan data surveyor
surveyor = pd.DataFrame(df,
                        columns=['Nama Surveyor (Mahasiswa )', 'NIM Surveyor (Mahasiswa)', 'Kelas Surveyor (Mahasiswa)', 'Unnamed: 3'])

# Melakukan drop terhadap kolom dan baris yang tidak dibutuhkan
surveyor['Kelas Surveyor (Mahasiswa)'].fillna(surveyor['Unnamed: 3'], inplace = True)
del surveyor['Unnamed: 3']
surveyor.dropna()
surveyor = surveyor.drop(0)

# Mengganti data dengan value yang lebih mudah dipahami
surveyor['Kelas Surveyor (Mahasiswa)'] = surveyor['Kelas Surveyor (Mahasiswa)'].map({1: 'STI', 2: 'EL'})

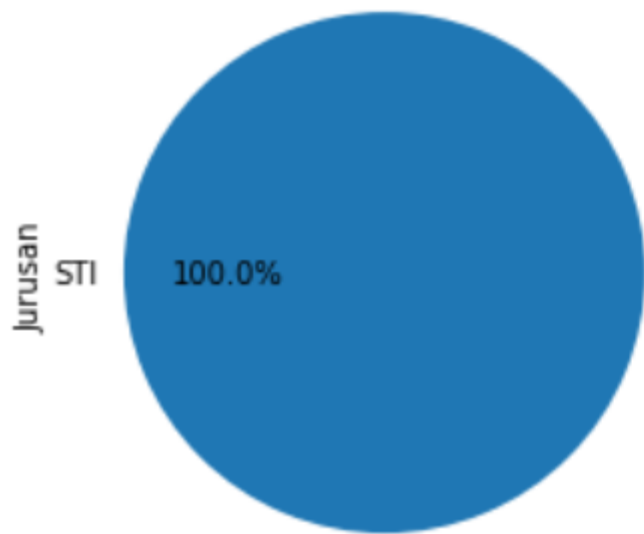
# Mengganti nama kolom
surveyor.rename(columns = {surveyor.columns[0] : 'Nama'}, inplace = True)
surveyor.rename(columns = {surveyor.columns[1] : 'NIM'}, inplace = True)
surveyor.rename(columns = {surveyor.columns[2] : 'Jurusan'}, inplace = True)
```

Lanjutan Mengolah Data Surveyor untuk Data Cleansing

```
# Melakukan sort untuk mencari data yang salah kemudian menghapusnya  
# Data yang dihapus seperti data dengan Nama: Test, j, a, iji, dan semacamnya  
# dan NIM yang bukan 18219000-18219118  
surveyor.NIM = surveyor.NIM.astype(str)  
surveyor = surveyor.sort_values(by=['NIM'], ascending = False)  
pd.set_option('display.max_rows', None)  
# display(surveyor) untuk mencari data yang salah  
surveyor = surveyor.drop([225,220,22,19,244,143,163,321,261])  
  
# Membuang duplicate dan data yang terisi salah  
surveyor = surveyor.drop_duplicates(subset=['NIM'])  
surveyor = surveyor.drop([94])
```

```
surveyor['Jurusan'].value_counts().plot(kind='pie', autopct='%1.1f%%')  
# Hasilnya 100% STI sehingga terbukti pembersihan data berjalan sukses  
# (Karena survey ini memang khusus untuk surveyor STI K1)
```

<AxesSubplot:ylabel='Jurusan'>



Data Cleansing

MENGHAPUS DATA YANG SUDAH PASTI TIDAK VALID

```
# Semua data yang dihapus ini sudah dicek di file excel untuk memastikan data tersebut memang "kurang baik"
# Menghapus data yang dalam 1 baris semua value-nya NaN
df = df.dropna(how='all')

# Menghapus data yang benar-benar salah
# Data yang dihapus seperti data dengan Nama Surveyor: Test, j, a, iji, dan semacamnya
# dan NIM Surveyor yang bukan 18219000-18219118, dengan pengecualian NIM yang hanya typo 1 angka tidak dihapus
df = df.drop([225,220,22,19,244,143,163,321])

# Data yang dihapus seperti data dengan Nama Responden: NaN, test, dan semacamnya
responden = pd.DataFrame(df,
                          columns=['Nama Responden'])
responden = responden.sort_values(by=['Nama Responden'], ascending = False)
# display(responden)
df = df.drop([323, 190, 285])
```

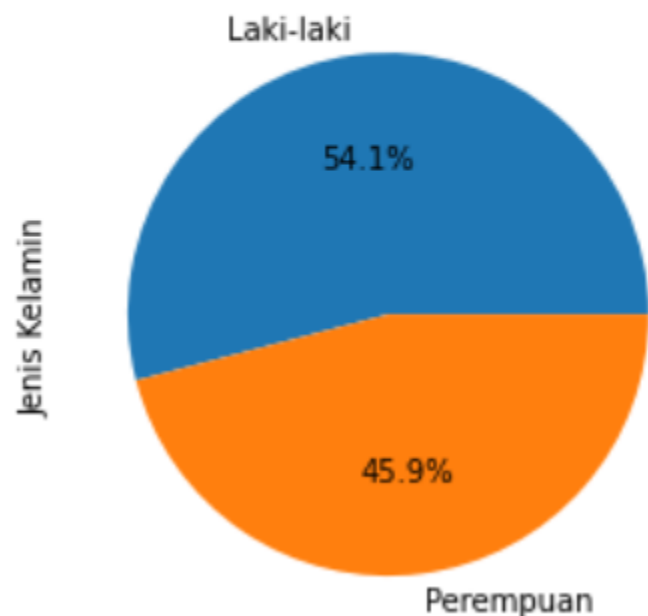
Latar Belakang Responden

DATA RESPONDEN

```
# Merapikan data jenisKelamin
jenisKelamin = pd.DataFrame(df,
                             columns=['Jenis kelamin ?', 'Unnamed: 6'])
jenisKelamin['Jenis kelamin ?'].fillna(jenisKelamin['Unnamed: 6'], inplace = True)
del jenisKelamin['Unnamed: 6']
jenisKelamin = jenisKelamin.dropna()
jenisKelamin = jenisKelamin.drop(0)
jenisKelamin['Jenis kelamin ?'] = jenisKelamin['Jenis kelamin ?'].map({1: 'Laki-laki', 2: 'Perempuan'})
jenisKelamin.rename(columns = {jenisKelamin.columns[0] : 'Jenis Kelamin'}, inplace = True)
# jenisKelamin.isnull().sum() hasilnya 0
# display(jenisKelamin)
```

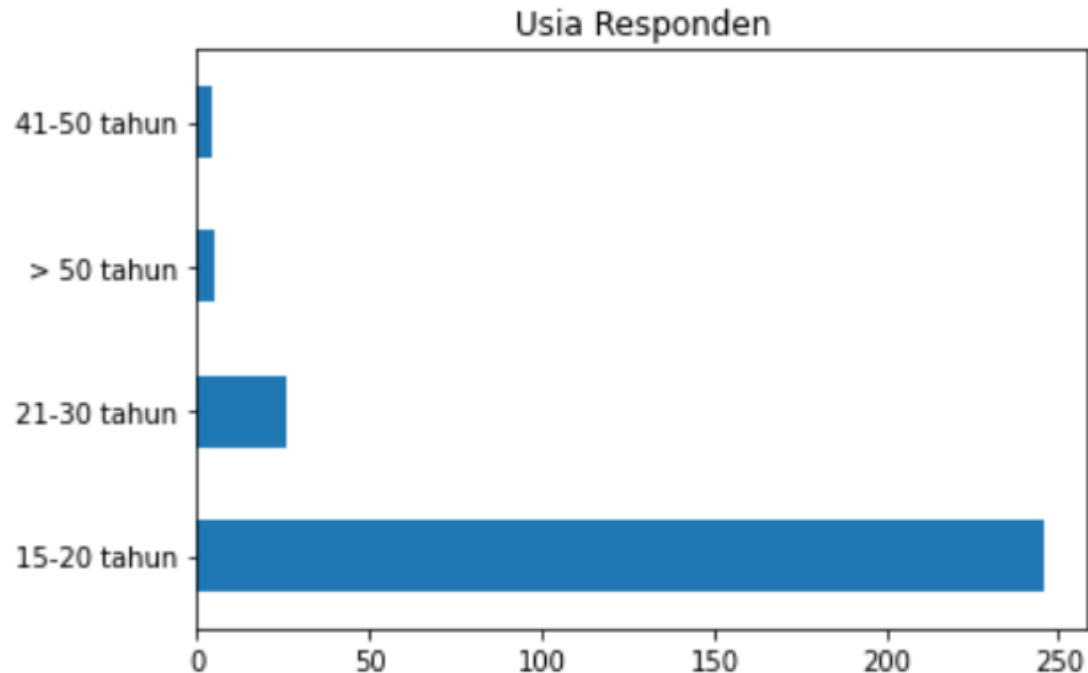
```
# Menampilkan jenis kelamin responden yang valid
jenisKelamin['Jenis Kelamin'].value_counts().plot(kind='pie', autopct='%1.1f%%')
```

```
<AxesSubplot:ylabel='Jenis Kelamin'>
```



Dengan cara yang sama bisa didapat :

```
<AxesSubplot:title={'center':'Usia Responden'}>
```



Analisis Aplikasi

APLIKASI YANG BIASA DIGUNAKAN UNTUK WORK FROM HOME/SCHOOL FROM HOME

```
app = pd.DataFrame(df,
    columns=['Aplikasi apa yang biasa digunakan untuk Work From Home/ School from Home ?', 'Unnamed: 186', 'Unnamed: 187',
        'Unnamed: 188', 'Unnamed: 189', 'Unnamed: 190', 'Unnamed: 191'])

app.rename(columns = {app.columns[0] : 'Aplikasi Biasa Digunakan Untuk WFH/SFH'}, inplace = True)

app['Aplikasi Biasa Digunakan Untuk WFH/SFH'] = app['Aplikasi Biasa Digunakan Untuk WFH/SFH'].map({1:'Whatsapp'})
app['Unnamed: 186'] = app['Unnamed: 186'].map({2:'Zoom'})
app['Unnamed: 187'] = app['Unnamed: 187'].map({3:'Google Meet'})
app['Unnamed: 188'] = app['Unnamed: 188'].map({4:'Microsoft Team'})
app['Unnamed: 189'] = app['Unnamed: 189'].map({5:'Cisco Webex'})
app['Unnamed: 190'] = app['Unnamed: 190'].map({6:'Skype for Business'})
app['Unnamed: 191'] = app['Unnamed: 191'].map({7:'GoToMeeting'})
```

Lanjutan Analisis Aplikasi

```
app['Aplikasi Biasa Digunakan Untuk WFH/SFH'] = app['Aplikasi Biasa Digunakan Untuk WFH/SFH'].replace([np.nan], ' ')
app['Unnamed: 186'] = app['Unnamed: 186'].replace([np.nan], ' ')
app['Unnamed: 187'] = app['Unnamed: 187'].replace([np.nan], ' ')
app['Unnamed: 188'] = app['Unnamed: 188'].replace([np.nan], ' ')
app['Unnamed: 189'] = app['Unnamed: 189'].replace([np.nan], ' ')
app['Unnamed: 190'] = app['Unnamed: 190'].replace([np.nan], ' ')
app['Unnamed: 191'] = app['Unnamed: 191'].replace([np.nan], ' ')

app['Aplikasi Biasa Digunakan Untuk WFH/SFH'] = app['Aplikasi Biasa Digunakan Untuk WFH/SFH'].map(str) + ' ' + app['Unnamed: 186']
del app['Unnamed: 186'], app['Unnamed: 187'], app['Unnamed: 188'], app['Unnamed: 189'], app['Unnamed: 190'], app['Unnamed: 191']
app = app.dropna(how='all')
app = app.drop(0)
```

N.B. Kode yang terpotong adalah sebagai berikut :

```
app['Aplikasi Biasa Digunakan Untuk WFH/SFH'] = app['Aplikasi Biasa Digunakan Untuk
WFH/SFH'].map(str) + ' ' + app['Unnamed: 186'].map(str) + ' ' + app['Unnamed: 187'].map(str) + '
' + app['Unnamed: 188'].map(str) + ' ' + app['Unnamed: 189'].map(str) + ' ' + app['Unnamed:
190'].map(str) + ' ' + app['Unnamed: 191'].map(str)
```

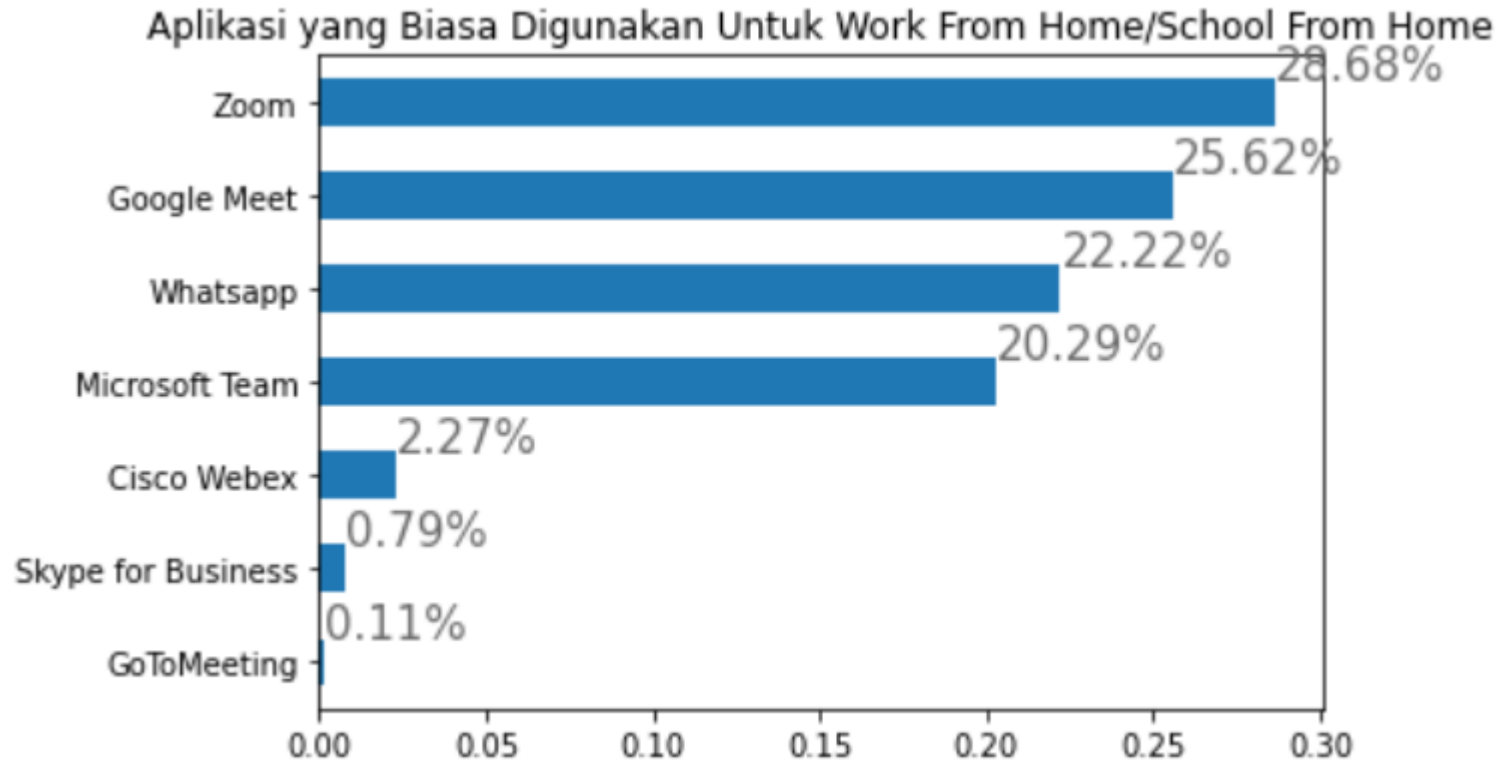
Lanjutan Analisis Aplikasi

```
apph = app['Aplikasi Biasa Digunakan Untuk WFH/SFH'].str.extractall(f'({"|".join(["Whatsapp", "Zoom", "Google Meet", "Microsoft T  
total1 = []  
for i in apph.patches:  
    total1.append(i.get_width())  
  
total11 = sum(total1)  
  
for i in apph.patches:  
    apph.text(i.get_width(), i.get_y(), str(round((i.get_width()/total11)*100,2)) + '%', fontsize=15,color='dimgrey')  
  
apph.invert_yaxis()
```

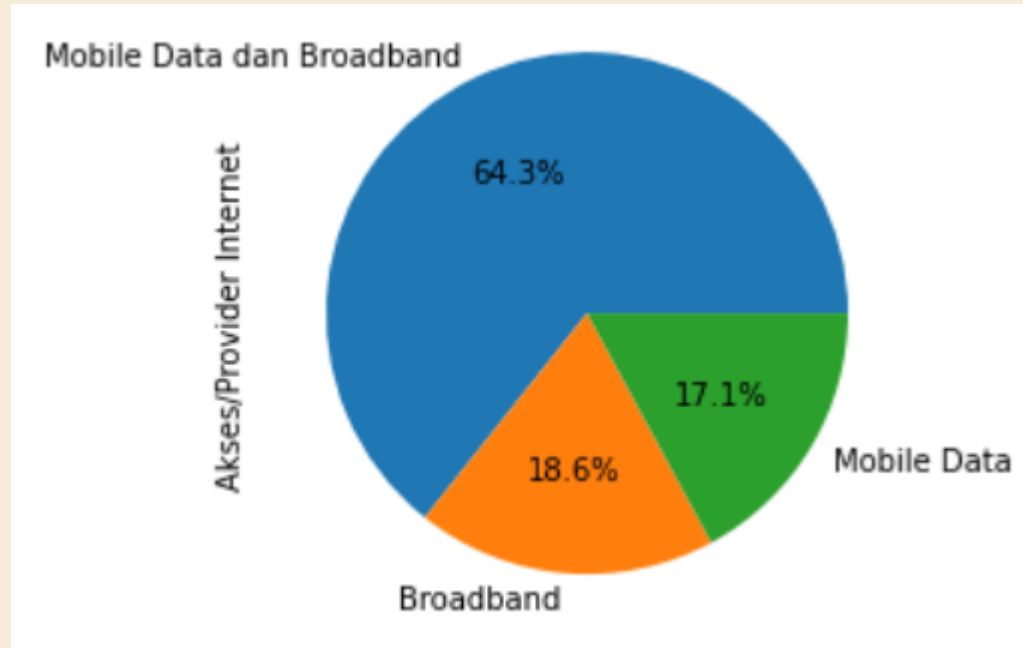
N.B. Kode yang terpotong adalah sebagai berikut :

```
apph = app['Aplikasi Biasa Digunakan Untuk WFH/SFH'].str.extractall(f'({"|".join(["Whatsapp",  
"Zoom", "Google Meet", "Microsoft Team", "Cisco Webex", "Skype for Business",  
"GoToMeeting"])}))')[0].value_counts('percent').plot.barh(title = 'Aplikasi yang Biasa Digunakan  
Untuk Work From Home/School From Home')
```

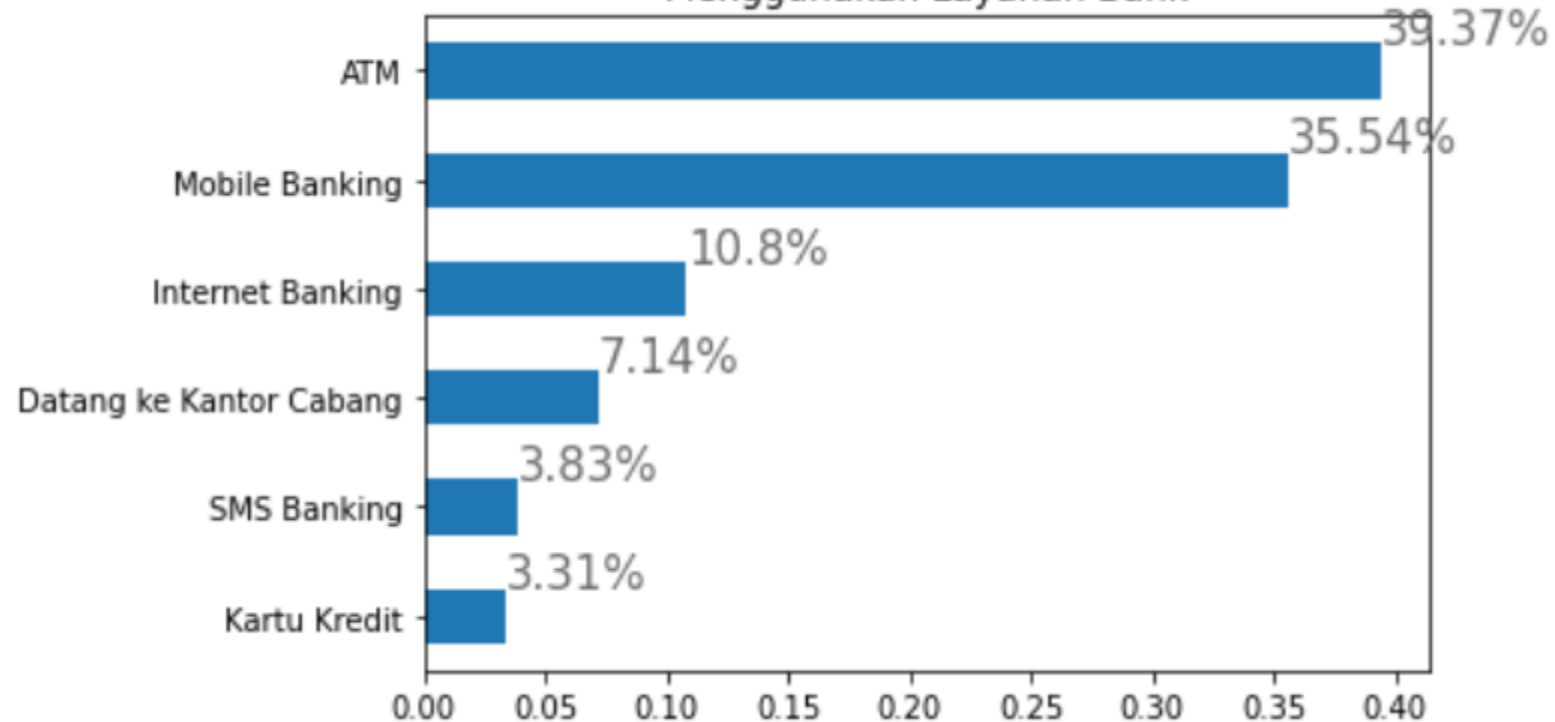
Hasil Analisis Aplikasi

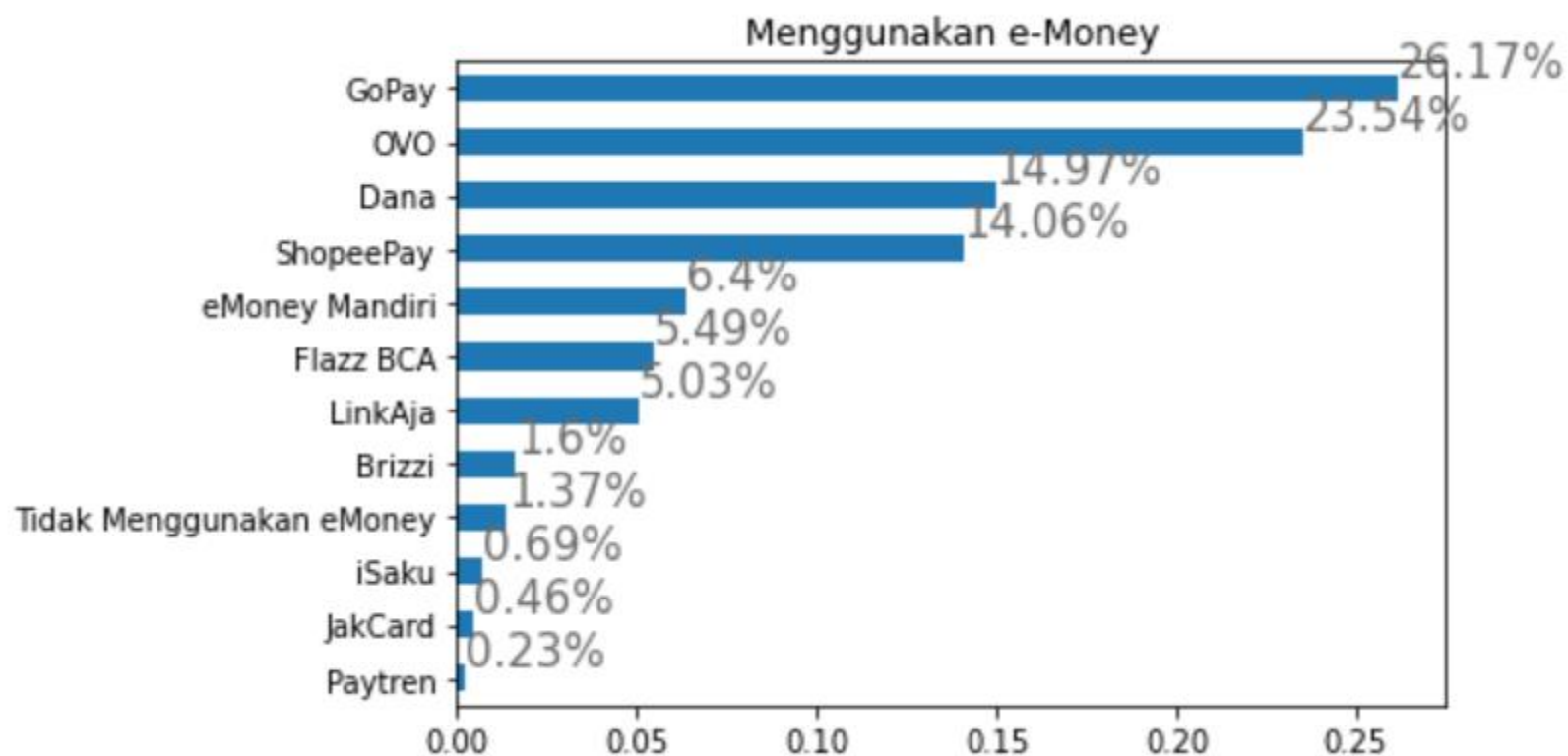


Dengan Cara yang Sama Didapat :

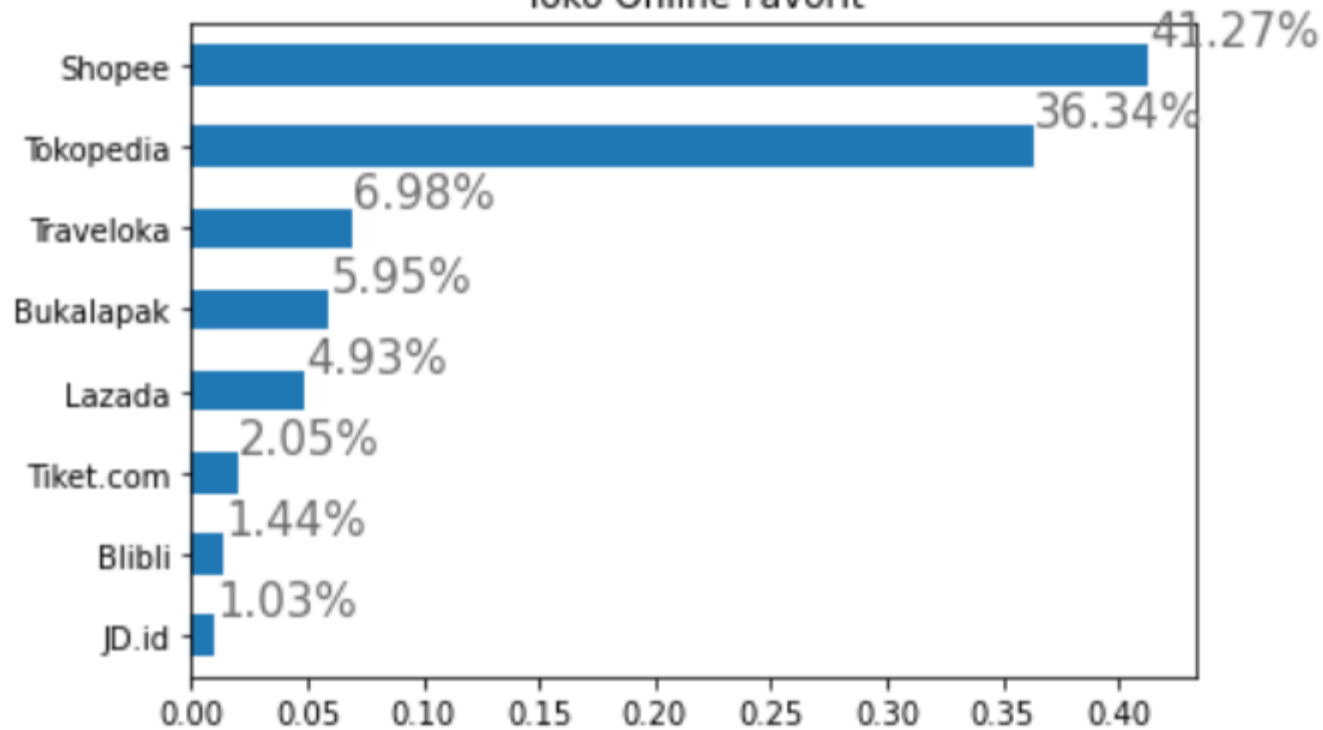


Menggunakan Layanan Bank





Toko Online Favorit

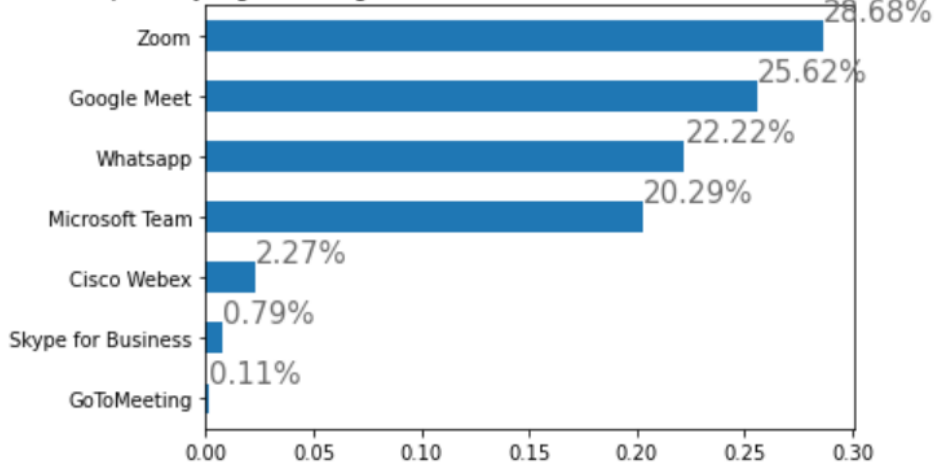


```
<AxesSubplot:title={'center':'Menggunakan Fintech'}, ylabel='0'>
```



Uji Statistik

Aplikasi yang Biasa Digunakan Untuk Work From Home/School From Home



n	p	q	z 0,05	bawah	atas	x	z uji
30	0,2868	0,7132	1,645	0,150968	0,422632	8,604	2,098416
100	0,2868	0,7132	1,645	0,212402	0,361198	28,68	3,831166
273	0,2868	0,7132	1,645	0,241772	0,331828	78,2964	6,330126

Aplikasi yang Biasa Digunakan untuk WFH/SFH

Hipotesis:

H0 : Masyarakat yang biasa menggunakan Zoom untuk WFH/SFH = 15%

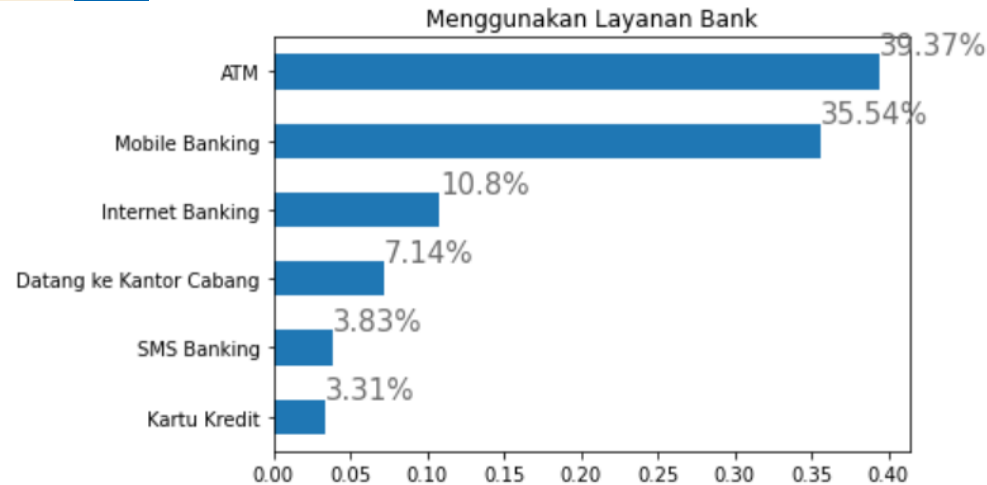
H1 : Masyarakat yang biasa menggunakan Zoom untuk WFH/SFH > 15%

Karena didapat Z uji > Z kritis, maka H0 ditolak, H1 diterima,

yang berarti terdapat cukup bukti yang menunjukkan bahwa

masyarakat yang biasa menggunakan Zoom untuk WFH/SFW lebih banyak dari 15%

Uji Statistik



n	p	q	z 0,05	bawah	atas	x	z uji
30	0,3937	0,6063	1,645	0,246966	0,540434	11,811	1,817677
100	0,3937	0,6063	1,645	0,31333	0,47407	39,37	3,318609
269	0,3937	0,6063	1,645	0,344698	0,442702	105,9053	5,442924

Layanan Bank yang Digunakan Masyarakat

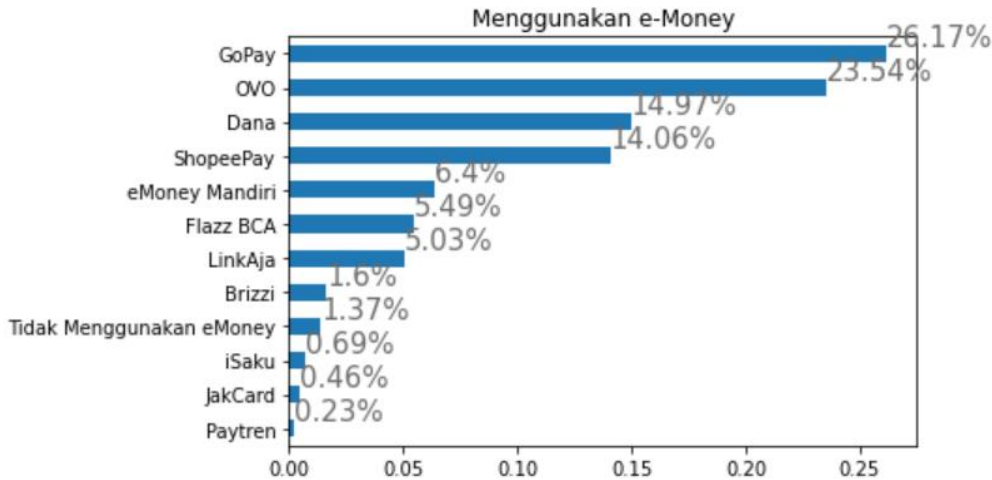
Hipotesis:

H_0 : Masyarakat yang menggunakan layanan bank ATM = 25%

H_1 : Masyarakat yang menggunakan layanan bank ATM > 25%

Karena didapat $Z_{uji} > Z_{kritis}$, maka H_0 ditolak, H_1 diterima, yang berarti terdapat cukup bukti yang menunjukkan bahwa masyarakat yang menggunakan layanan bank ATM lebih banyak dari 25%

Uji Statistik



n	p	q	z 0,05	bawah	atas	x	z uji
30	0,2617	0,7383	1,645	0,129685	0,393715	7,851	1,7134
100	0,2617	0,7383	1,645	0,189392	0,334008	26,17	3,128226
254	0,2617	0,7383	1,645	0,21633	0,30707	66,4718	4,985571

e-Money yang Digunakan Masyarakat

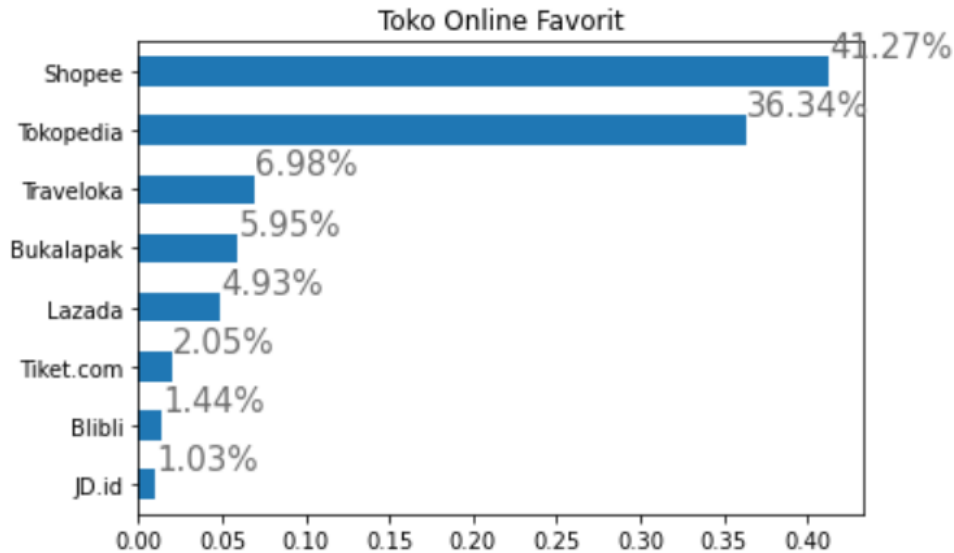
Hipotesis:

H0 : Masyarakat yang menggunakan e-money GoPay = 15%

H1 : Masyarakat yang menggunakan e-money GoPay > 15%

Karena didapat Z uji > Z kritis, maka H0 ditolak, H1 diterima, yang berarti terdapat cukup bukti yang menunjukkan bahwa masyarakat yang menggunakan e-money Gopay lebih banyak dari 15%

Uji Statistik



n	p	q	z 0,05	bawah	atas	x	z uji
30	0,3634	0,6366	1,645	0,218946	0,507854	10,902	1,815326
100	0,3634	0,6366	1,645	0,284279	0,442521	36,34	3,314317
257	0,3634	0,6366	1,645	0,314046	0,412754	93,3938	5,313254

Toko Online Favorit Masyarakat

Hipotesis:

H0 : Masyarakat yang toko online favoritnya Tokopedia = 22,5%

H1 : Masyarakat yang toko online favoritnya Tokopedia > 22,5%

Karena didapat Z uji > Z kritis, maka H0 ditolak, H1 diterima,
yang berarti terdapat cukup bukti yang menunjukkan bahwa
masyarakat yang toko online favoritnya Tokopedia lebih banyak dari 22,5%



**TERIMA
KASIH**