Data Exploration:

In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:

```
# Read the dataset
df = pd.read_csv(r'F:\Technocolabs\WA_Fn-UseC_-HR-Employee-Attrition.csv')
```

In [3]:

```
df
```

Out[3]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education
0	41	Yes	Travel_Rarely	1102	Sales	1	2
1	49	No	Travel_Frequently	279	Research & Development	8	1
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2
3	33	No	Travel_Frequently	1392	Research & Development	3	4
4	27	No	Travel_Rarely	591	Research & Development	2	1
...	...	...	...	...	...	...	...
1465	36	No	Travel_Frequently	884	Research & Development	23	2
1466	39	No	Travel_Rarely	613	Research & Development	6	1
1467	27	No	Travel_Rarely	155	Research & Development	4	3
1468	49	No	Travel_Frequently	1023	Sales	2	3
1469	34	No	Travel_Rarely	628	Research & Development	8	3

1470 rows × 35 columns

In [4]:
```python
# Display first few rows
print(df.head())
```

```
   Age Attrition     BusinessTravel  DailyRate              Department  \
0   41       Yes      Travel_Rarely       1102                   Sales   
1   49        No  Travel_Frequently        279  Research & Development   
2   37       Yes      Travel_Rarely       1373  Research & Development   
3   33        No  Travel_Frequently       1392  Research & Development   
4   27        No      Travel_Rarely        591  Research & Development   

   DistanceFromHome  Education EducationField  EmployeeCount  EmployeeNumb
er  \
0                 1          2  Life Sciences              1              
1   
1                 8          1  Life Sciences              1              
2   
2                 2          2          Other              1              
4   
3                 3          4  Life Sciences              1              
5   
4                 2          1        Medical              1              
7   

   ...  RelationshipSatisfaction StandardHours  StockOptionLevel  \
0  ...                         1            80                 0   
1  ...                         4            80                 1   
2  ...                         2            80                 0   
3  ...                         3            80                 0   
4  ...                         4            80                 1   

   TotalWorkingYears  TrainingTimesLastYear WorkLifeBalance  YearsAtCompan
y  \
0                  8                      0               1              
6   
1                 10                      3               3              1
0   
2                  7                      3               3              
0   
3                  8                      3               3              
8   
4                  6                      3               3              
2   

  YearsInCurrentRole  YearsSinceLastPromotion  YearsWithCurrManager  
0                  4                        0                     5  
1                  7                        1                     7  
2                  0                        0                     0  
3                  7                        3                     0  
4                  2                        2                     2  

[5 rows x 35 columns]
```

In [5]:

```
# Summary statistics
print(df.describe())
```

```
               Age    DailyRate  DistanceFromHome    Education  EmployeeCo
unt  \
count  1470.000000  1470.000000       1470.000000  1470.000000          147
0.0
mean     36.923810   802.485714          9.192517     2.912925
1.0
std       9.135373   403.509100          8.106864     1.024165
0.0
min      18.000000   102.000000          1.000000     1.000000
1.0
25%      30.000000   465.000000          2.000000     2.000000
1.0
50%      36.000000   802.000000          7.000000     3.000000
1.0
75%      43.000000  1157.000000         14.000000     4.000000
1.0
max      60.000000  1499.000000         29.000000     5.000000
1.0

       EmployeeNumber  EnvironmentSatisfaction   HourlyRate  JobInvolvemen
t  \
count     1470.000000              1470.000000  1470.000000     1470.00000
0
mean      1024.865306                 2.721769    65.891156        2.72993
2
std        602.024335                 1.093082    20.329428        0.71156
1
min          1.000000                 1.000000    30.000000        1.00000
0
25%        491.250000                 2.000000    48.000000        2.00000
0
50%       1020.500000                 3.000000    66.000000        3.00000
0
75%       1555.750000                 4.000000    83.750000        3.00000
0
max       2068.000000                 4.000000   100.000000        4.00000
0

          JobLevel  ...  RelationshipSatisfaction  StandardHours  \
count  1470.000000  ...               1470.000000         1470.0
mean      2.063946  ...                  2.712245           80.0
std       1.106940  ...                  1.081209            0.0
min       1.000000  ...                  1.000000           80.0
25%       1.000000  ...                  2.000000           80.0
50%       2.000000  ...                  3.000000           80.0
75%       3.000000  ...                  4.000000           80.0
max       5.000000  ...                  4.000000           80.0

       StockOptionLevel  TotalWorkingYears  TrainingTimesLastYear  \
count       1470.000000        1470.000000            1470.000000
mean           0.793878          11.279592               2.799320
std            0.852077           7.780782               1.289271
min            0.000000           0.000000               0.000000
25%            0.000000           6.000000               2.000000
50%            1.000000          10.000000               3.000000
75%            1.000000          15.000000               3.000000
max            3.000000          40.000000               6.000000

       WorkLifeBalance  YearsAtCompany  YearsInCurrentRole  \
count      1470.000000     1470.000000         1470.000000
mean          2.761224        7.008163            4.229252
```

```
std          0.706476       6.126525            3.623137
min          1.000000       0.000000            0.000000
25%          2.000000       3.000000            2.000000
50%          3.000000       5.000000            3.000000
75%          3.000000       9.000000            7.000000
max          4.000000      40.000000           18.000000

       YearsSinceLastPromotion  YearsWithCurrManager
count              1470.000000           1470.000000
mean                  2.187755              4.123129
std                   3.222430              3.568136
min                   0.000000              0.000000
25%                   0.000000              2.000000
50%                   1.000000              3.000000
75%                   3.000000              7.000000
max                  15.000000             17.000000

[8 rows x 26 columns]
```

In [7]:
```python
# Value counts for categorical variables
print(df['MonthlyRate'].value_counts())
```

```
4223     3
9150     3
9558     2
12858    2
22074    2
        ..
14561    1
2671     1
5718     1
11757    1
10228    1
Name: MonthlyRate, Length: 1427, dtype: int64
```

In [8]:
```python
print(df['DailyRate'].value_counts())
```

```
691     6
408     5
530     5
1329    5
1082    5
       ..
650     1
279     1
316     1
314     1
628     1
Name: DailyRate, Length: 886, dtype: int64
```

In [9]:
```python
print(df['Attrition'].value_counts())
```

```
No     1233
Yes     237
Name: Attrition, dtype: int64
```

In [10]:
```
print(df['BusinessTravel'].value_counts())
```

```
Travel_Rarely        1043
Travel_Frequently     277
Non-Travel            150
Name: BusinessTravel, dtype: int64
```

In [11]:
```
print(df['Department'].value_counts())
```

```
Research & Development    961
Sales                     446
Human Resources            63
Name: Department, dtype: int64
```

In [12]:
```
print(df['EducationField'].value_counts())
```

```
Life Sciences       606
Medical             464
Marketing           159
Technical Degree    132
Other                82
Human Resources      27
Name: EducationField, dtype: int64
```

In [13]:
```
print(df['Gender'].value_counts())
```

```
Male      882
Female    588
Name: Gender, dtype: int64
```

In [14]:
```
print(df['JobRole'].value_counts())
```

```
Sales Executive              326
Research Scientist           292
Laboratory Technician        259
Manufacturing Director       145
Healthcare Representative    131
Manager                      102
Sales Representative          83
Research Director             80
Human Resources               52
Name: JobRole, dtype: int64
```

In [15]:
```
print(df['MaritalStatus'].value_counts())
```

```
Married     673
Single      470
Divorced    327
Name: MaritalStatus, dtype: int64
```

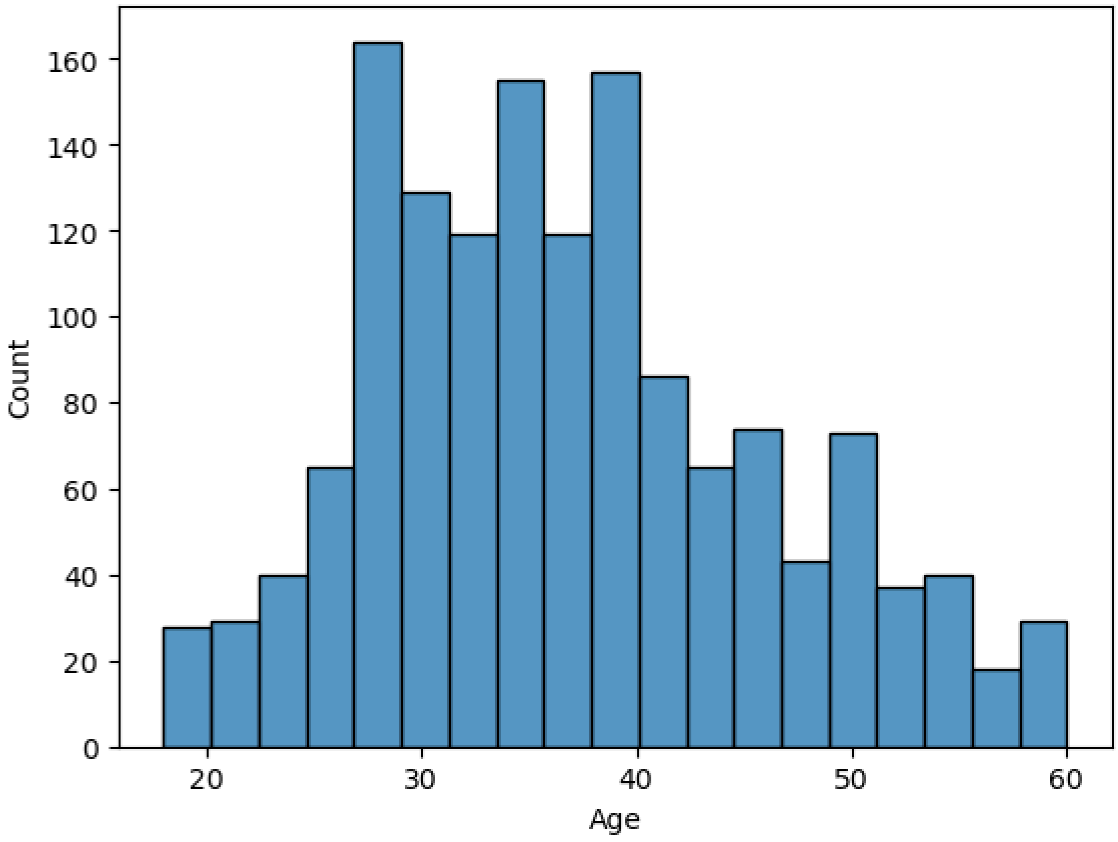
In [16]:
```
print(df['Over18'].value_counts())
```

```
Y    1470
Name: Over18, dtype: int64
```

In [17]:
```
print(df['OverTime'].value_counts())
```

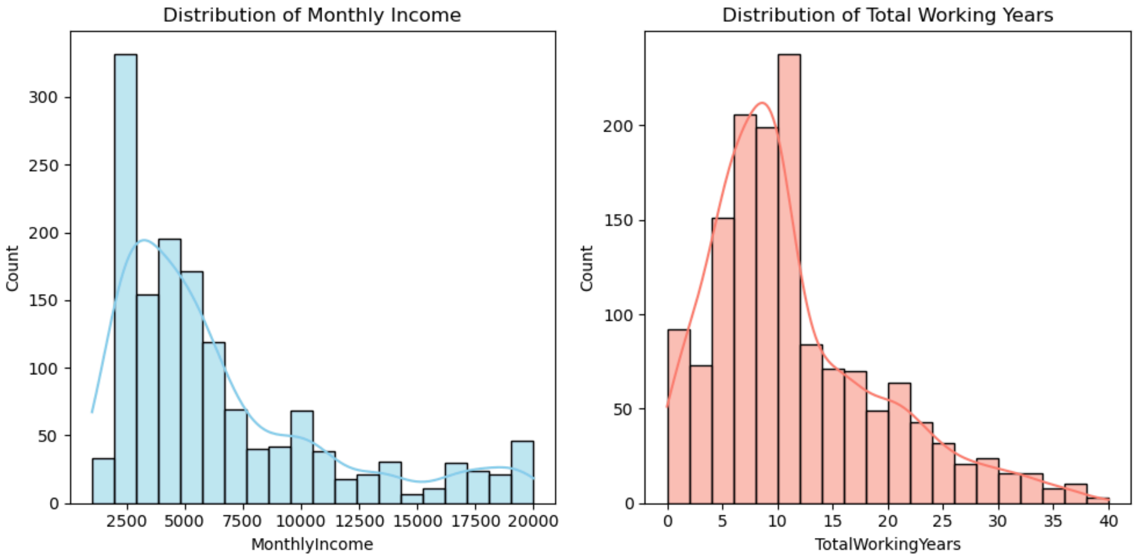
```
No     1054
Yes     416
Name: OverTime, dtype: int64
```

In [18]:
```
# Visualization
sns.histplot(df['Age'])
plt.show()
```



In [22]:

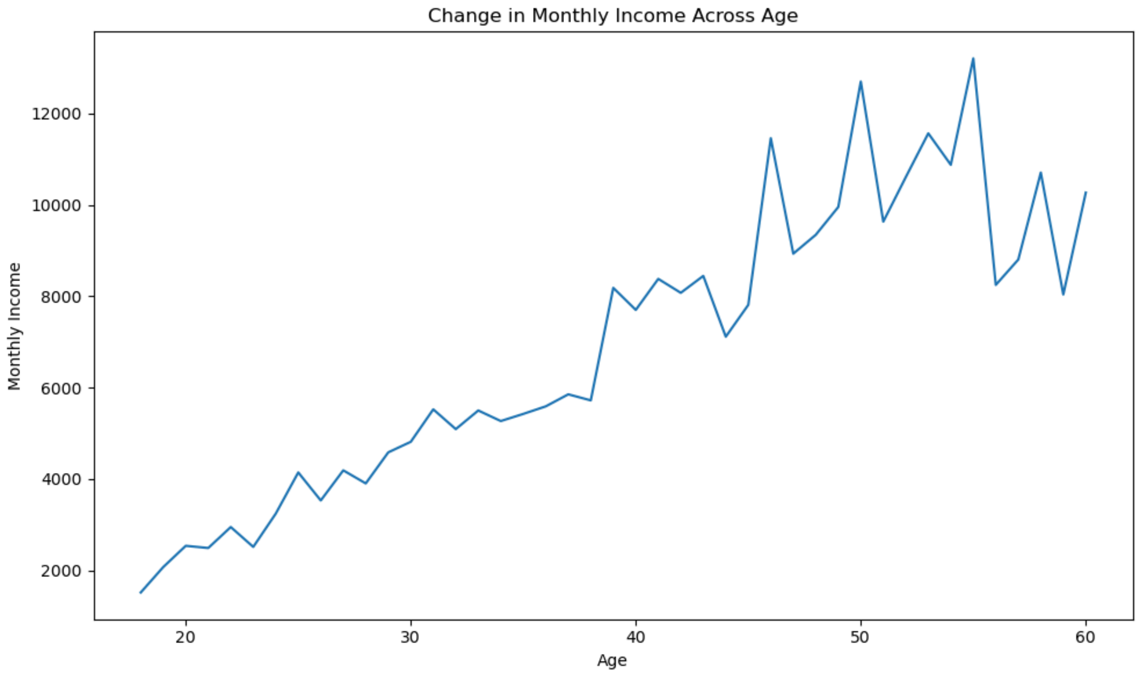
```
import seaborn as sns
import matplotlib.pyplot as plt

# Visualization
plt.figure(figsize=(10, 5))

# Histogram for Monthly Income
plt.subplot(1, 2, 1)
sns.histplot(df['MonthlyIncome'], bins=20, kde=True, color='skyblue')
plt.title('Distribution of Monthly Income')

# Histogram for Total Working Years
plt.subplot(1, 2, 2)
sns.histplot(df['TotalWorkingYears'], bins=20, kde=True, color='salmon')
plt.title('Distribution of Total Working Years')

plt.tight_layout()
plt.show()
```



In [21]:

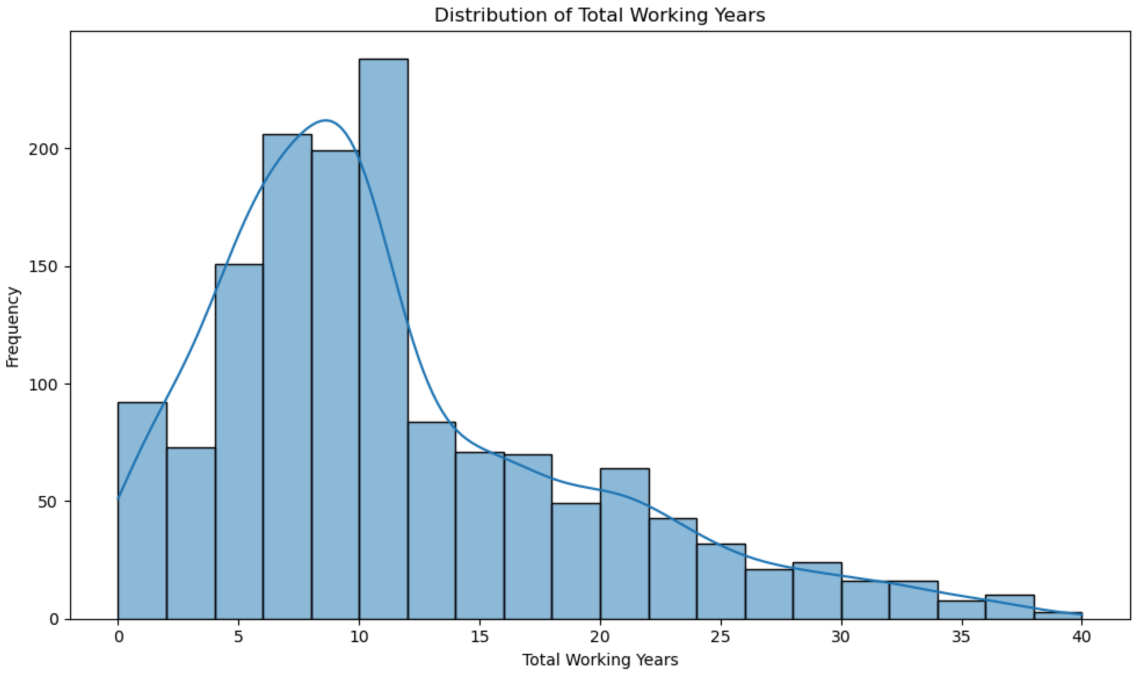
```python
# Visualization
plt.figure(figsize=(10, 6))

# Line plot for change in Monthly Income across Age
sns.lineplot(x='Age', y='MonthlyIncome', data=df, ci=None)
plt.title('Change in Monthly Income Across Age')
plt.xlabel('Age')
plt.ylabel('Monthly Income')

plt.tight_layout()
plt.show()
```

```
C:\Temp2\ipykernel_1364\2436244171.py:5: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

  sns.lineplot(x='Age', y='MonthlyIncome', data=df, ci=None)
```



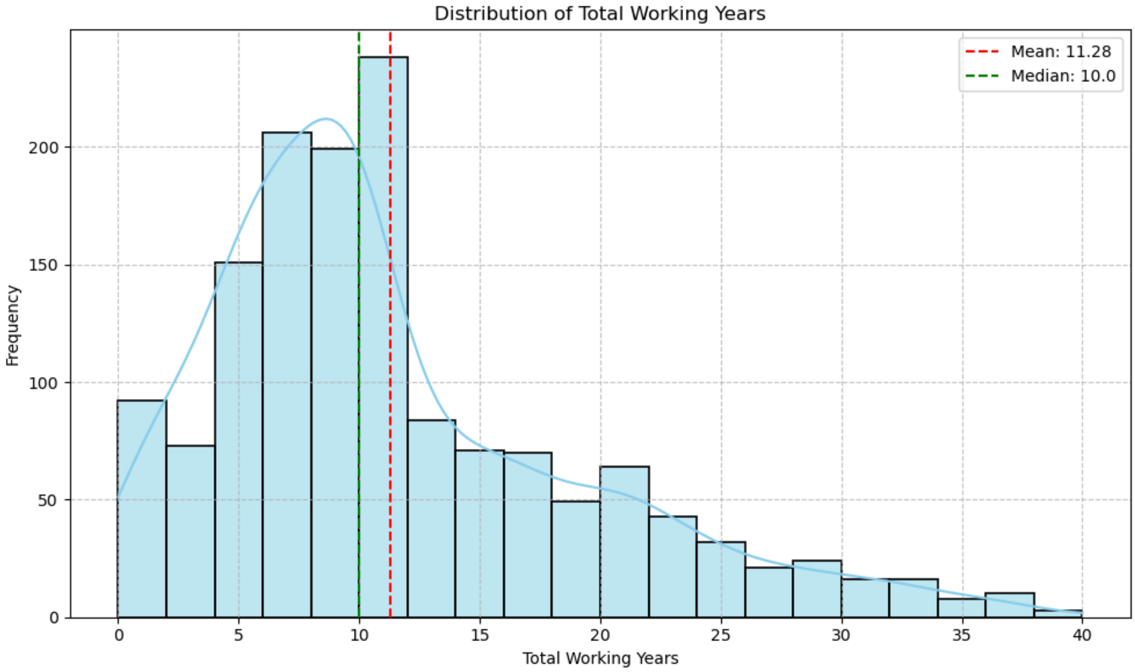
In [23]:

```
# Visualization
plt.figure(figsize=(10, 6))

# Histogram for Total Working Years
sns.histplot(df['TotalWorkingYears'], bins=20, kde=True)
plt.title('Distribution of Total Working Years')
plt.xlabel('Total Working Years')
plt.ylabel('Frequency')

plt.tight_layout()
plt.show()
```



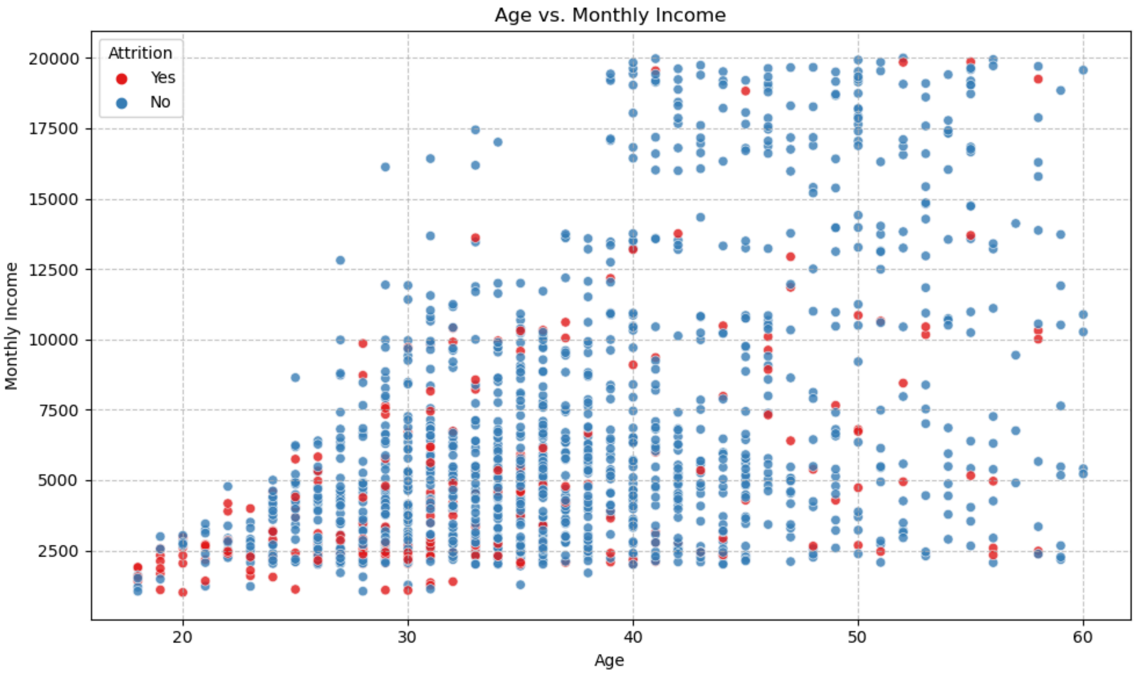
In [24]:

```
import seaborn as sns
import matplotlib.pyplot as plt

# Visualization
plt.figure(figsize=(10, 6))

# Histogram for Total Working Years
sns.histplot(df['TotalWorkingYears'], bins=20, kde=True, color='skyblue', e
plt.title('Distribution of Total Working Years')
plt.xlabel('Total Working Years')
plt.ylabel('Frequency')

# Adding grid for better readability
plt.grid(True, linestyle='--', alpha=0.7)

# Adding mean and median lines
mean_total_working_years = df['TotalWorkingYears'].mean()
median_total_working_years = df['TotalWorkingYears'].median()
plt.axvline(mean_total_working_years, color='red', linestyle='--', label=f'
plt.axvline(median_total_working_years, color='green', linestyle='--', labe

# Adding legend
plt.legend()

plt.tight_layout()
plt.show()
```



In [25]:

```
# Additional Visualization
plt.figure(figsize=(10, 6))

# Scatter plot: Age vs. Monthly Income
sns.scatterplot(data=df, x='Age', y='MonthlyIncome', hue='Attrition', palet
plt.title('Age vs. Monthly Income')
plt.xlabel('Age')
plt.ylabel('Monthly Income')

# Adding grid for better readability
plt.grid(True, linestyle='--', alpha=0.7)

# Adding legend
plt.legend(title='Attrition')

plt.tight_layout()
plt.show()
```



In [ ]: