In [ ]:
```
#Trevor Zeiger
#DSC - 680
#Week3 Milestone 2
```

In [ ]:
```
# Remote Work Salary Analysis - Data Cleaning and Combining Script (with Detailed E
```

In [33]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
```

In [35]:
```
warnings.filterwarnings('ignore')  # Suppress warnings for cleaner output
```

In [37]:
```
# ----------------------------
# Load all datasets
# ----------------------------
# These are the three datasets used in this analysis:
# - ds_salaries.csv: Global dataset containing remote work salary data across indus
# - eda_data.csv: A U.S.-focused dataset derived from Glassdoor job postings
# - Salary_Dataset_with_Extra_Features.csv: An India-based dataset, largely focused

# In a real implementation, replace the file paths with your own local or cloud-bas

ds_salaries = pd.read_csv("ds_salaries.csv")
eda_data = pd.read_csv("eda_data.csv")
salary_dataset = pd.read_csv("Salary_Dataset_with_Extra_Features.csv")
```

In [39]:
```
# Display the first few rows of each dataset to understand their structure
ds_salaries_head = ds_salaries.head()
eda_data_head = eda_data.head()
salary_dataset_head = salary_dataset.head()

ds_salaries_head, eda_data_head, salary_dataset_head
```

Out[39]: (Unnamed: 0 work_year experience_level employment_type \
0   0   2020   MI   FT
1   1   2020   SE   FT
2   2   2020   SE   FT
3   3   2020   MI   FT
4   4   2020   SE   FT

                    job_title salary salary_currency salary_in_usd \
0   Data Scientist   70000   EUR   79833
1 Machine Learning Scientist 260000   USD   260000
2   Big Data Engineer   85000   GBP   109024
3   Product Data Analyst   20000   USD   20000
4   Machine Learning Engineer 150000   USD   150000

  employee_residence remote_ratio company_location company_size
0   DE   0   DE   L
1   JP   0   JP   S
2   GB   50   GB   M
3   HN   0   HN   S
4   US   50   US   L ,
   Unnamed: 0   Job Title   Salary Estimate \
0   0   Data Scientist $53K-$91K (Glassdoor est.)
1   1 Healthcare Data Scientist $63K-$112K (Glassdoor est.)
2   2   Data Scientist $80K-$90K (Glassdoor est.)
3   3   Data Scientist $56K-$97K (Glassdoor est.)
4   4   Data Scientist $86K-$143K (Glassdoor est.)

                                     Job Description Rating \
0 Data Scientist\nLocation: Albuquerque, NM\nEdu...   3.8
1 What You Will Do:\n\nI. General Summary\n\nThe...   3.4
2 KnowBe4, Inc. is a high growth information sec...   4.8
3 *Organization and Job ID**\nJob ID: 310709\n\n...   3.8
4 Data Scientist\nAffinity Solutions / Marketing...   2.9

                                Company Name   Location \
0   Tecolote Research\n3.8 Albuquerque, NM
1 University of Maryland Medical System\n3.4   Linthicum, MD
2   KnowBe4\n4.8   Clearwater, FL
3   PNNL\n3.8   Richland, WA
4   Affinity Solutions\n2.9   New York, NY

     Headquarters   Size Founded ... age python_yn R_yn \
0   Goleta, CA   501 to 1000 employees   1973 ... 47   1   0
1   Baltimore, MD   10000+ employees   1984 ... 36   1   0
2 Clearwater, FL   501 to 1000 employees   2010 ... 10   1   0
3   Richland, WA 1001 to 5000 employees   1965 ... 55   1   0
4   New York, NY   51 to 200 employees   1998 ... 22   1   0

  spark aws excel   job_simp seniority desc_len num_comp
0   0   0   1 data scientist   na   2536   0
1   0   0   0 data scientist   na   4783   0
2   1   0   1 data scientist   na   3461   0
3   0   0   0 data scientist   na   3883   3
4   0   0   1 data scientist   na   2728   3

[5 rows x 33 columns],

```
   Rating                    Company Name          Job Title   Salary  \
0     3.8                          Sasken  Android Developer   400000
1     4.5  Advanced Millennium Technologies  Android Developer   400000
2     4.0                       Unacademy  Android Developer  1000000
3     3.8             SnapBizz Cloudtech  Android Developer   300000
4     4.4          Appoids Tech Solutions  Android Developer   600000

   Salaries Reported   Location Employment Status Job Roles
0                  3  Bangalore         Full Time   Android
1                  3  Bangalore         Full Time   Android
2                  3  Bangalore         Full Time   Android
3                  3  Bangalore         Full Time   Android
4                  3  Bangalore         Full Time   Android  )
```

In [41]:
```
# ----------------------------
# Define a consistent column structure
# ----------------------------
# To combine different datasets effectively, we define a set of standard column nam
# that will be used across all datasets, even if some values are missing or estimat
standard_columns = [
    'Job Title',              # Name or type of the position (e.g., Data Scientist)
    'Location',               # Geographic location or company base
    'Employment Type',        # Full-time, part-time, contract, etc.
    'Experience Level',       # Entry, mid, senior, executive (may be missing in so
    'Salary (USD)',           # Salary converted to USD where possible
    'Salary (INR)',           # Salaries specific to India-based roles (in Indian R
    'Salary Estimate',        # Text-based salary range estimates from sources like
    'Source'                  # Indicates the dataset origin: Global, Glassdoor, or
]
```

In [43]:
```
#----------------------------
# Clean and reformat each dataset
# ----------------------------

# --- Global dataset (ds_salaries.csv) ---
# This dataset already includes structured salary data and location info.
ds_clean = pd.DataFrame(columns=standard_columns)
ds_clean['Job Title'] = ds_salaries['job_title']
ds_clean['Location'] = ds_salaries['company_location']
ds_clean['Employment Type'] = ds_salaries['employment_type']
ds_clean['Experience Level'] = ds_salaries['experience_level']
ds_clean['Salary (USD)'] = ds_salaries['salary_in_usd']
ds_clean['Source'] = 'Global'
```

In [45]:
```
# --- Glassdoor dataset (eda_data.csv) ---
# This dataset is rich in job descriptions and salary estimates but lacks structure
eda_clean = pd.DataFrame(columns=standard_columns)
eda_clean['Job Title'] = eda_data['job_simp']                    # Simplified job ti
eda_clean['Location'] = eda_data['Location']                    # U.S. city/state in
eda_clean['Salary Estimate'] = eda_data['Salary Estimate']      # Salary range text
eda_clean['Source'] = 'Glassdoor'
```

In [47]:
```
# --- Indeed India dataset (Salary_Dataset_with_Extra_Features.csv) ---
# This is a more localized dataset with salary data mostly in INR for Indian roles.
salary_clean = pd.DataFrame(columns=standard_columns)
```

```
salary_clean['Job Title'] = salary_dataset['Job Title']
salary_clean['Location'] = salary_dataset['Location']
salary_clean['Salary (INR)'] = salary_dataset['Salary']
salary_clean['Employment Type'] = salary_dataset['Employment Status']
salary_clean['Source'] = 'India'
```

In [49]:
```
# ----------------------------
# Combine all cleaned datasets
# ----------------------------
# We concatenate the three datasets into a single DataFrame for unified analysis.
# Missing values are expected for some columns depending on the source.
combined_df = pd.concat([ds_clean, eda_clean, salary_clean], ignore_index=True)
```

In [51]:
```
# ----------------------------
# Preview the combined dataset
# ----------------------------
# This output allows us to confirm the structure and integrity of the merged datase
print(combined_df.head())
```

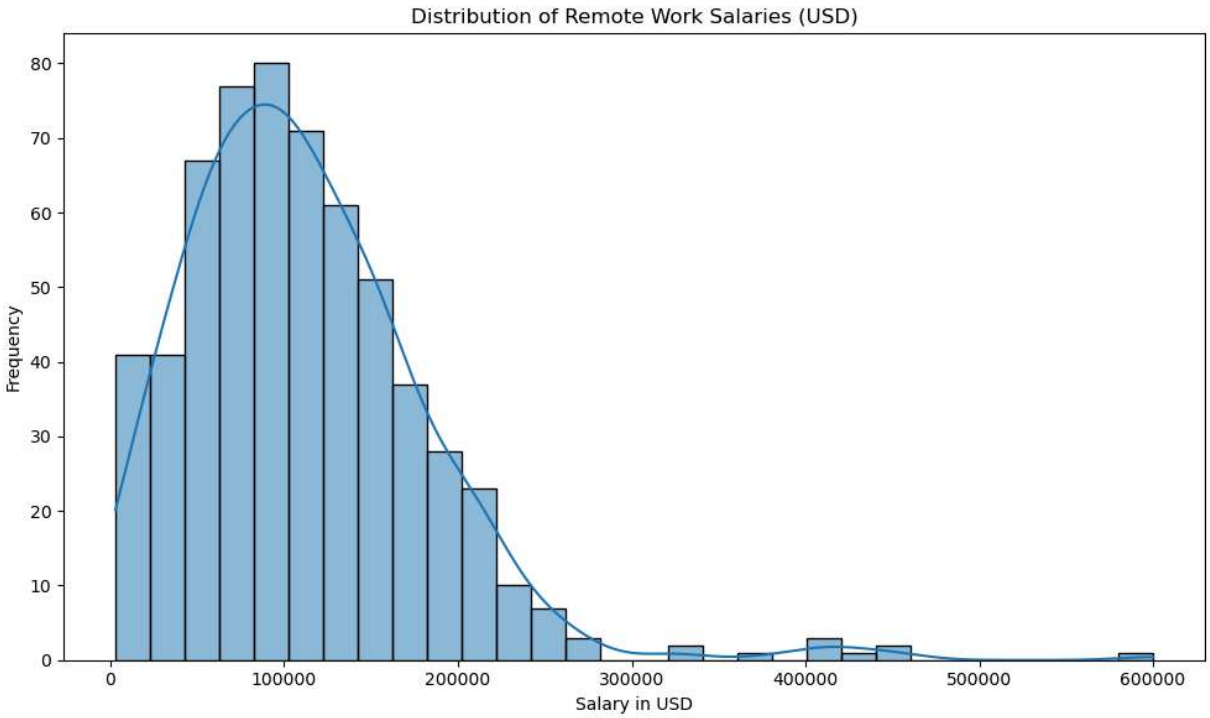
```
                   Job Title Location Employment Type Experience Level  \
0             Data Scientist       DE              FT               MI
1  Machine Learning Scientist       JP              FT               SE
2          Big Data Engineer       GB              FT               SE
3       Product Data Analyst       HN              FT               MI
4   Machine Learning Engineer       US              FT               SE

  Salary (USD) Salary (INR) Salary Estimate  Source
0        79833          NaN             NaN  Global
1       260000          NaN             NaN  Global
2       109024          NaN             NaN  Global
3        20000          NaN             NaN  Global
4       150000          NaN             NaN  Global
```

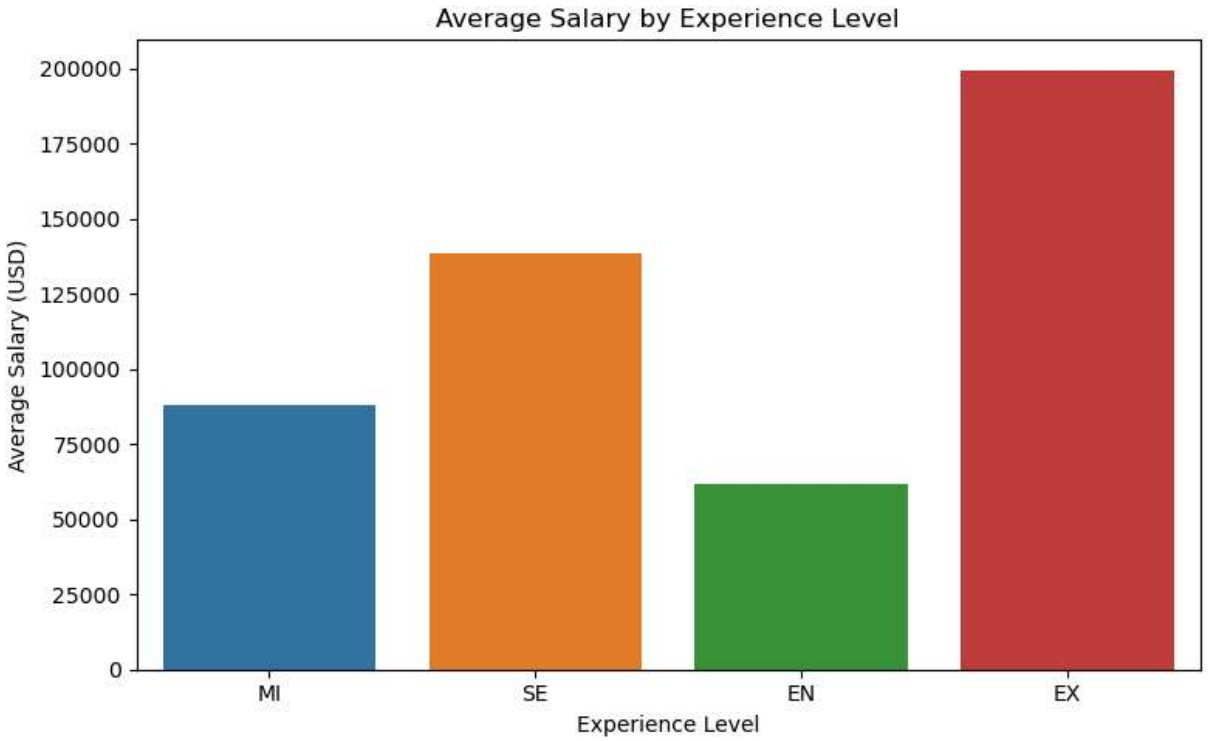
In [53]:
```
# ----------------------------
# Optional - Save the cleaned dataset
# ----------------------------
# This will create a CSV file for future analysis or visualization steps.
combined_df.to_csv("combined_remote_work_salary_data.csv", index=False)
```

In [55]:
```
# Filter out null salaries in USD for plotting
usd_data = combined_df[combined_df['Salary (USD)'].notnull()]
```

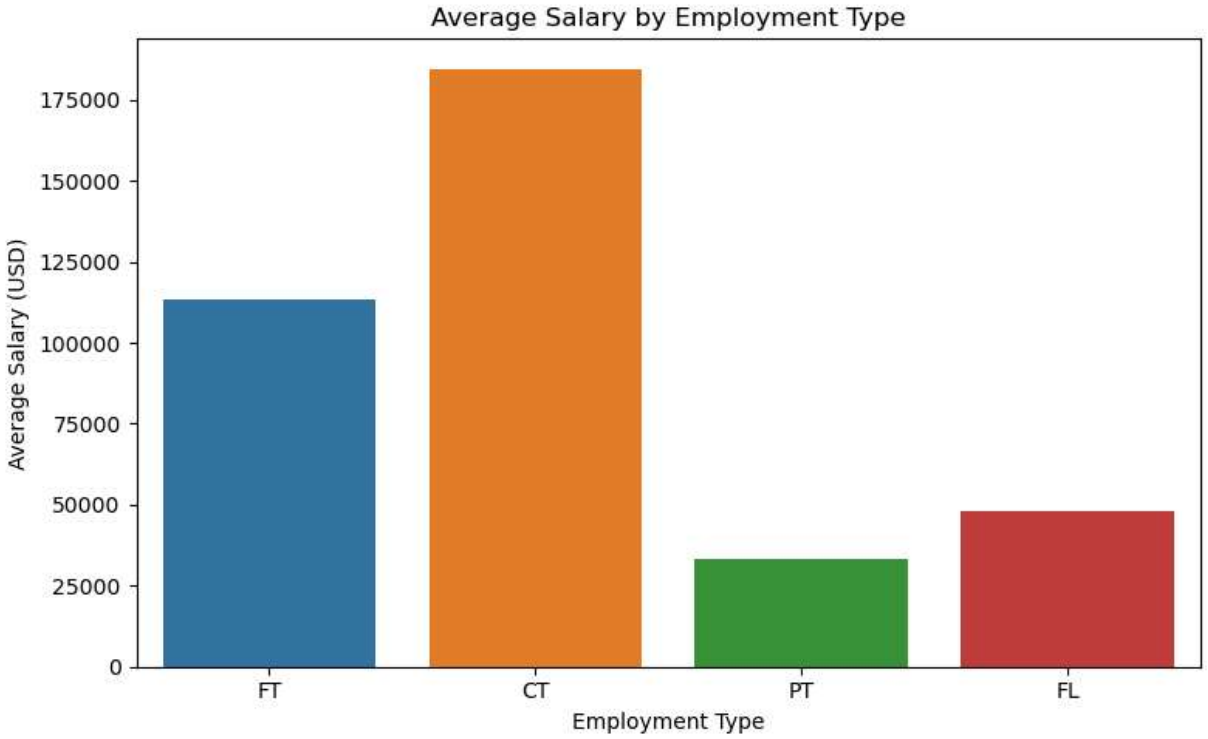
In [57]:
```
# Visualization 1: Distribution of salaries (USD)
plt.figure(figsize=(10, 6))
sns.histplot(usd_data['Salary (USD)'], bins=30, kde=True)
plt.title('Distribution of Remote Work Salaries (USD)')
plt.xlabel('Salary in USD')
plt.ylabel('Frequency')
plt.tight_layout()
plt.show()
```



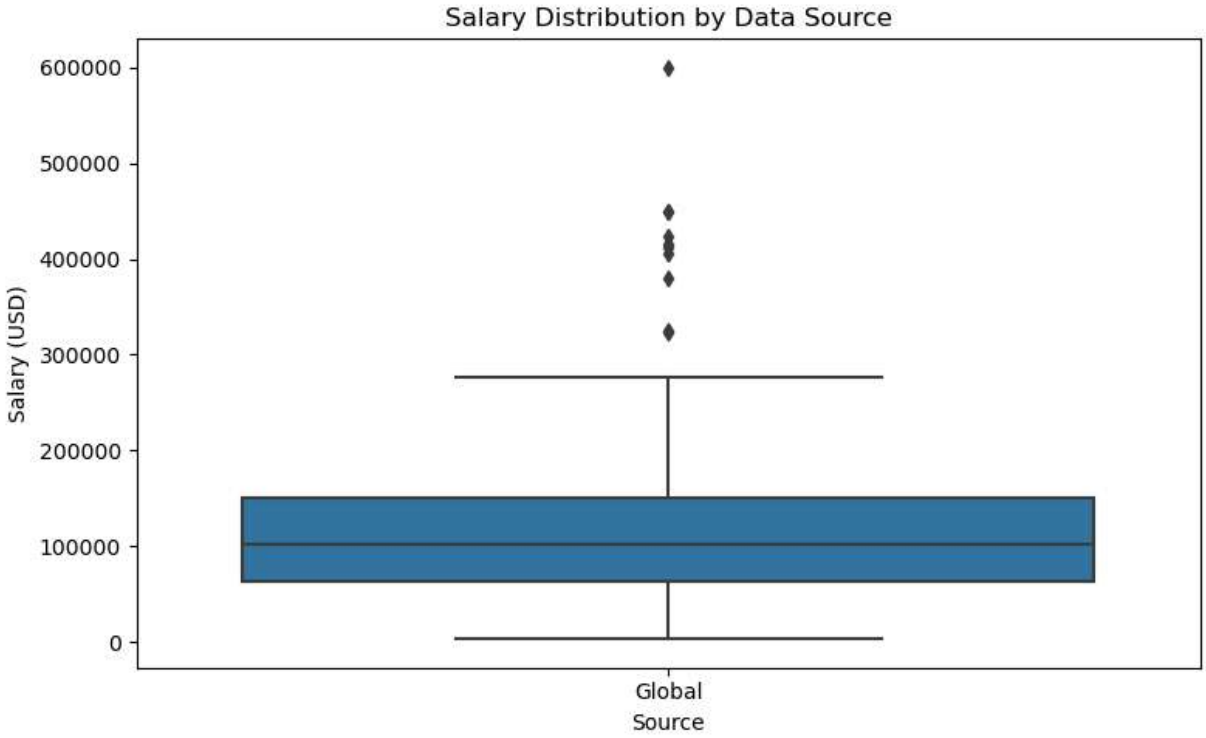
In [59]:
```
# Visualization 2: Average salary (USD) by Experience Level
plt.figure(figsize=(8, 5))
sns.barplot(data=usd_data, x='Experience Level', y='Salary (USD)', estimator='mean'
plt.title('Average Salary by Experience Level')
plt.xlabel('Experience Level')
plt.ylabel('Average Salary (USD)')
plt.tight_layout()
plt.show()
```



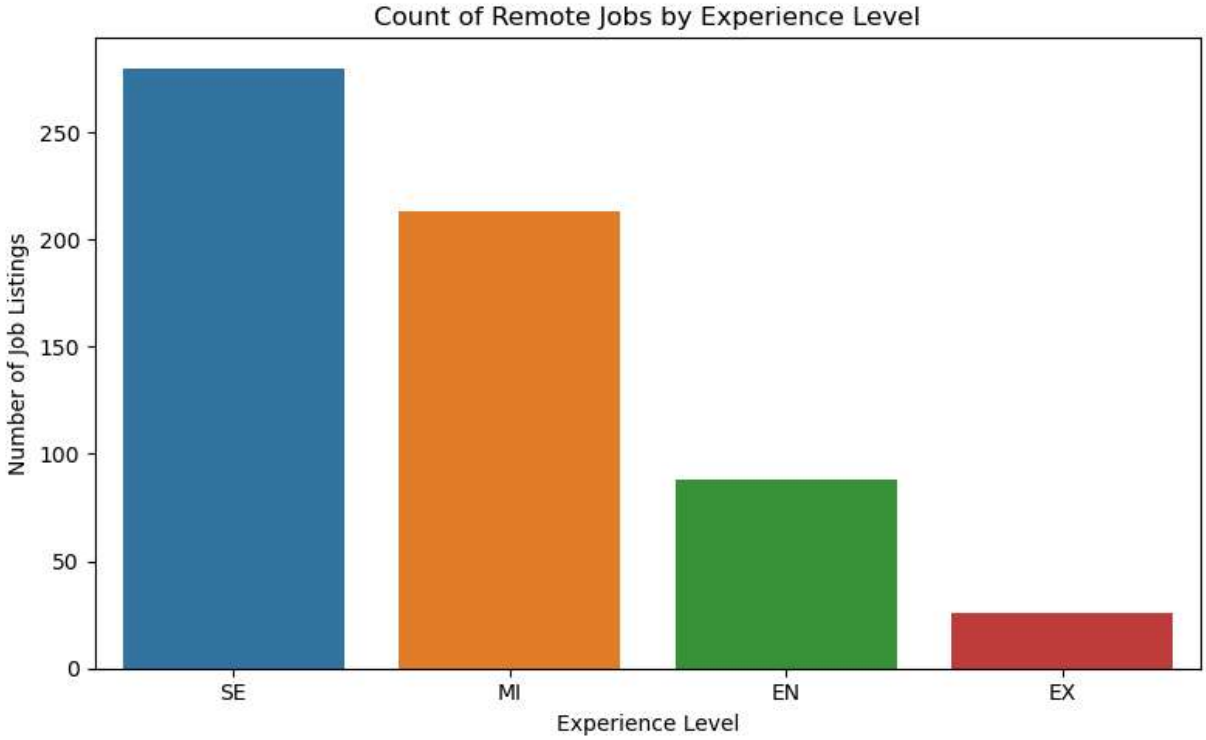
In [61]:
```
# Visualization 3: Average salary (USD) by Employment Type
plt.figure(figsize=(8, 5))
sns.barplot(data=usd_data, x='Employment Type', y='Salary (USD)', estimator='mean',
plt.title('Average Salary by Employment Type')
plt.xlabel('Employment Type')
plt.ylabel('Average Salary (USD)')
plt.tight_layout()
plt.show()
```



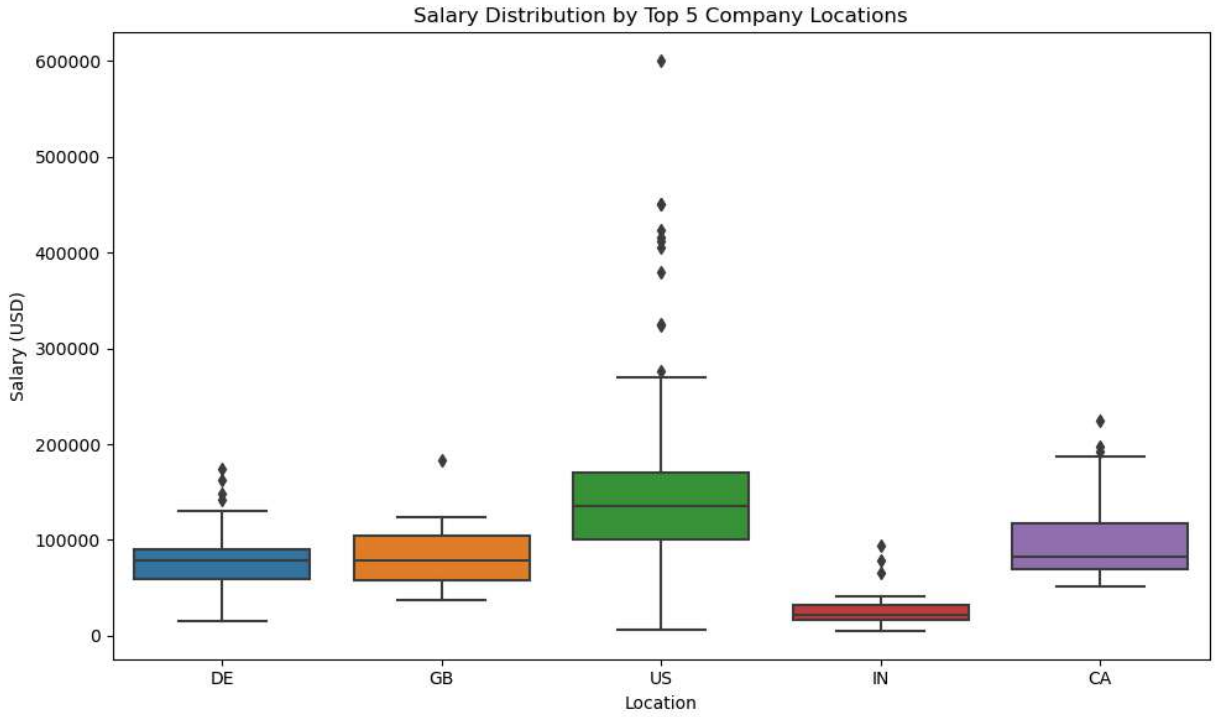
In [63]:
```
# Visualization 4: Salary comparison by Source
plt.figure(figsize=(8, 5))
sns.boxplot(data=usd_data, x='Source', y='Salary (USD)')
plt.title('Salary Distribution by Data Source')
plt.xlabel('Source')
plt.ylabel('Salary (USD)')
plt.tight_layout()
plt.show()
```



In [65]:
```
# Visualization 5: Count of Job Listings by Experience Level
plt.figure(figsize=(8, 5))
sns.countplot(data=usd_data, x='Experience Level', order=usd_data['Experience Level
plt.title('Count of Remote Jobs by Experience Level')
plt.xlabel('Experience Level')
plt.ylabel('Number of Job Listings')
plt.tight_layout()
plt.show()
```



In [67]:
```
# Visualization 6: Salary Trends by Company Location (Top 5)
top_locations = usd_data['Location'].value_counts().head(5).index
plt.figure(figsize=(10, 6))
sns.boxplot(data=usd_data[usd_data['Location'].isin(top_locations)], x='Location',
plt.title('Salary Distribution by Top 5 Company Locations')
plt.xlabel('Location')
plt.ylabel('Salary (USD)')
plt.tight_layout()
plt.show()
```



In [ ]: