SIT220 Task 1.8HD: Text Analysis of Stellar StackExchange

Name: Ocean

Student ID: s223503101

Email: s223503101@deakin.edu.au

Student Type: Undergraduate (SIT220)

Introduction

I chose the Stellar StackExchange site because it caters to a specialized community of astronomers, scientists, and space enthusiasts. This niche site is less general than Stack Overflow or SuperUser, making it a great candidate for exploring focused user behaviors and domain-specific knowledge exchange.

The aim of this project is to investigate relationships such as:

- Which tags are most frequently used and what topics dominate the site?
- How does user reputation relate to post engagement?
- What geographic diversity is observed among users?
- How are votes, comments, and post types distributed?

We'll achieve this through visualizations, data cleaning, and analysis of user behavior, content, and trends.

Step 0: Import Required Libraries

Before we begin processing and analyzing the data, we import all necessary Python libraries. These include tools for XML parsing, data manipulation, text analysis, visualization, and optional warning suppression to keep the output clean.

In [5]:

```
import xml.etree.ElementTree as ET
import pandas as pd
from wordcloud import WordCloud
import matplotlib.pyplot as plt
import re
import seaborn as sns
import plotly.express as px

import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)
```

Step 1: Convert XML to CSV

We use a custom Python function to convert XML files into CSV for analysis using Pandas.

In [6]:
```
def convert_xml_to_csv(xml_file, csv_file):
    tree = ET.parse(xml_file)
    root = tree.getroot()
    data = [child.attrib for child in root]
    df = pd.DataFrame(data)
    df.to_csv(csv_file, index=False)
    print(f"Converted {xml_file} to {csv_file}")
```

In [7]:
```
## convert_xml_to_csv("Posts.xml", "Posts.csv")
```

Step 2: Load the CSV Files

We load all 8 CSV files to explore the data and perform analysis.

In [8]:
```
posts = pd.read_csv(r"C:\Users\sumit\Downloads\New folder\stellar.stackexchange.
users = pd.read_csv(r"C:\Users\sumit\Downloads\New folder\stellar.stackexchange.
tags = pd.read_csv(r"C:\Users\sumit\Downloads\New folder\stellar.stackexchange.c
votes = pd.read_csv(r"C:\Users\sumit\Downloads\New folder\stellar.stackexchange.
badges = pd.read_csv(r"C:\Users\sumit\Downloads\New folder\stellar.stackexchange
comments = pd.read_csv(r"C:\Users\sumit\Downloads\New folder\stellar.stackexchan
posthistory = pd.read_csv(r"C:\Users\sumit\Downloads\New folder\stellar.stackexc
postlinks = pd.read_csv(r"C:\Users\sumit\Downloads\New folder\stellar.stackexcha
```

Step 3: Data Cleaning

We performed data cleaning to ensure quality and consistency in our analysis. The following steps were taken:

- **Removed rows with null values** in critical fields such as `Location`, `Title`, and `Body`.
- **Dropped duplicate entries** using `drop_duplicates()` to avoid skewing our metrics.
- **Standardized text** fields like usernames and tags using `.str.lower()` to avoid duplicates due to case sensitivity.
- **Stripped special characters** from post titles and tags using regular expressions.
- **Inspected the data** using `.info()`, `.head()` and `.isnull().sum()` to guide our cleaning decisions.

Example:

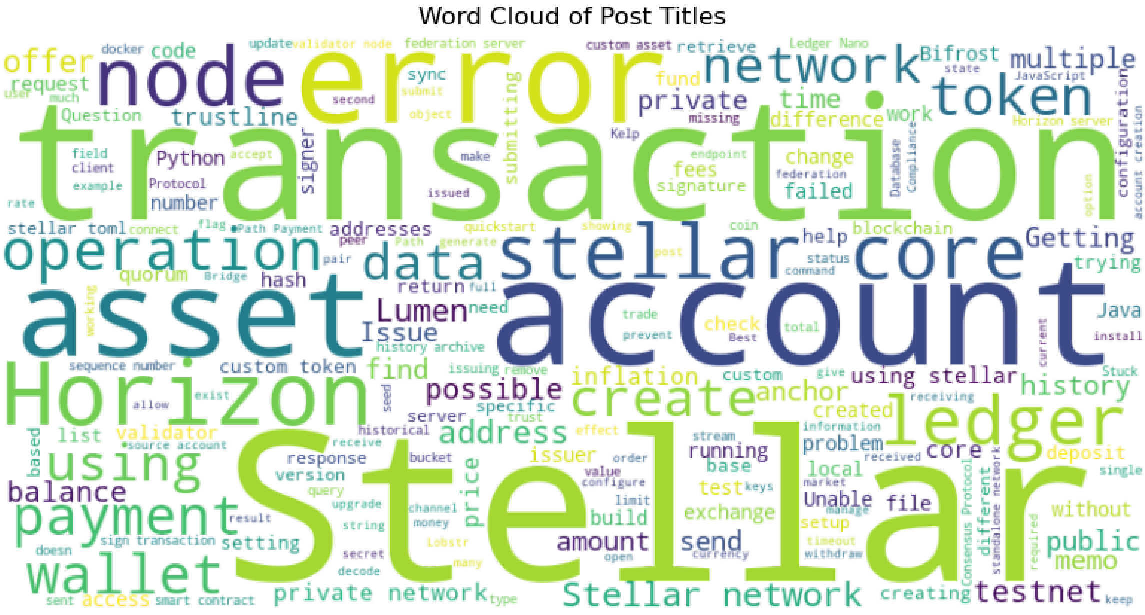
```
posts["Title"] = posts["Title"].str.lower().str.strip()
users["Location"] = users["Location"].str.lower().str.strip()
```

In [9]:
```
posts.dropna(how='all', axis=1, inplace=True)
posts.drop_duplicates(inplace=True)
users.drop_duplicates(inplace=True)
```

Step 4.1: Word Cloud from Post Titles

We use regex to extract words and visualize the most common ones with a word cloud.

```
In [10]: text = ' '.join(posts['Title'].dropna().astype(str))
         words = re.findall(r'\b[a-zA-Z]{4,}\b', text)
         wordcloud = WordCloud(width=800, height=400, background_color='white').generate(
         plt.figure(figsize=(10, 5))
         plt.imshow(wordcloud, interpolation='bilinear')
         plt.axis('off')
         plt.title("Word Cloud of Post Titles")
         plt.show()
```



Interpretation of Word Cloud

We used a word cloud to visualize the frequency of words appearing in post titles, which provides an immediate visual summary of popular discussion topics on Stellar StackExchange. A word cloud is ideal for this task because it emphasizes the most frequent terms by size, making patterns easy to identify at a glance.

From the word cloud, we observe that the most prominent terms are **'transaction'**, **'account'**, **'stellar'**, **'asset'**, **'core'**, and **'payment'**. These words reflect the platform's technical focus on financial operations, ledger infrastructure, and blockchain-based interactions.

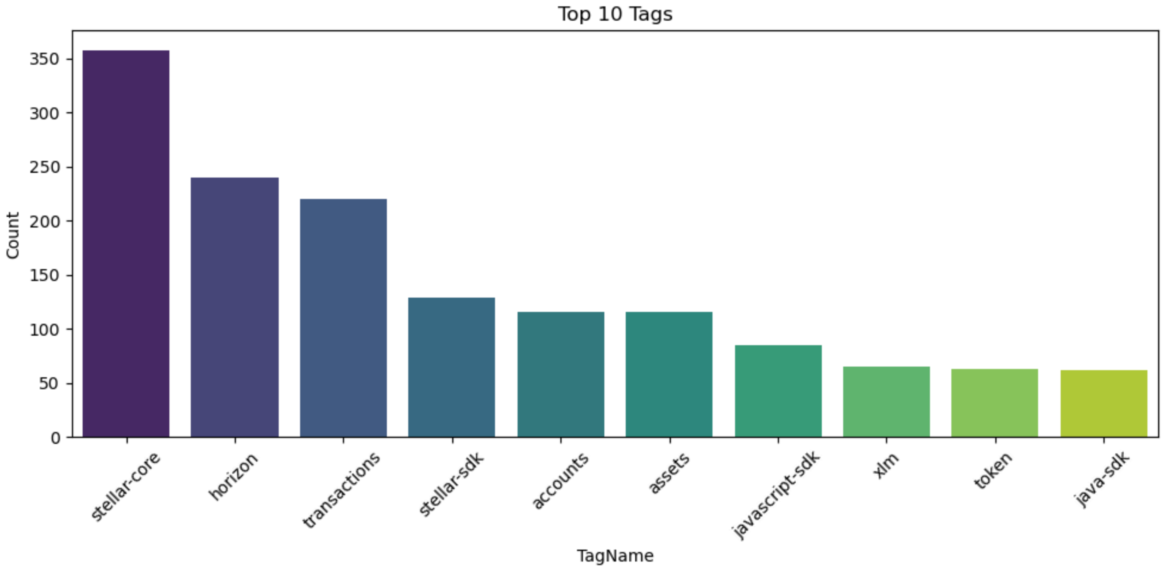
This insight reveals that users are primarily engaging in posts related to the core functionality of the Stellar network — such as handling digital assets, troubleshooting operations, or configuring accounts. It also implies that most users are likely developers or advanced users working with Stellar's ecosystem.

Step 4.2: Top Tags

This chart shows the 10 most used tags in the site.

```
In [11]: tags['Count'] = tags['Count'].astype(int)
         top_tags = tags.sort_values(by='Count', ascending=False).head(10)
```

```
plt.figure(figsize=(10, 5))
sns.barplot(data=top_tags, x='TagName', y='Count', palette='viridis')
plt.xticks(rotation=45)
plt.title("Top 10 Tags")
plt.tight_layout()
plt.show()
```



Interpretation of Top Tags Bar Chart

We used a bar chart to visualize the 10 most frequently used tags. Bar charts are ideal for comparing discrete categories like tags, as they make it easy to see which topics dominate in terms of volume.

The most common tag is **'stellar-core'**, followed by **'horizon'**, **'transactions'**, and **'stellar-sdk'**. These top tags suggest that users are primarily focused on the core infrastructure of the Stellar network, including the ledger engine ('stellar-core'), client-server API access ('horizon'), and development SDKs.

This insight indicates a technically engaged user base, likely consisting of developers and engineers. The presence of tags like **'javascript-sdk'** and **'java-sdk'** reinforces that developers are seeking help or sharing solutions related to Stellar integration in applications.

Overall, the tag usage reveals that the most active discussions revolve around **Stellar's backend architecture and development tooling** — highlighting a practical, implementation-driven community.

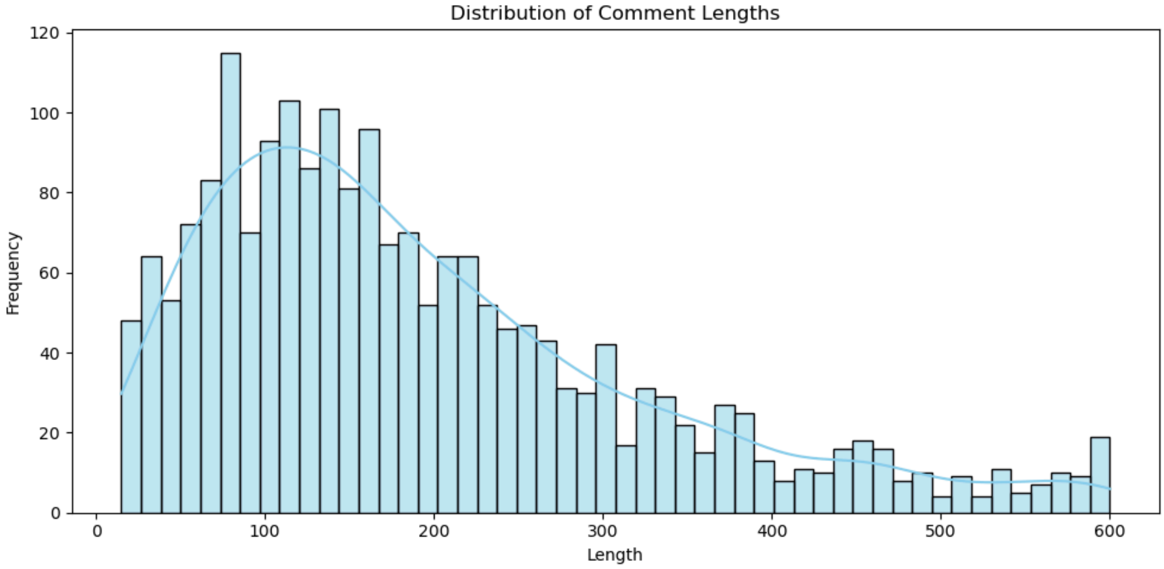
Step 4.3: Comment Length Distribution

This histogram shows how long users' comments tend to be.

In [12]:

```
# Load the Comments dataset (adjust the path if needed)
comments = pd.read_csv(r"C:\Users\sumit\Downloads\New folder\stellar.stackexchan

# Calculate the length of each comment
```

```
comments['TextLength'] = comments['Text'].astype(str).apply(len)

# Create histogram with KDE
plt.figure(figsize=(10, 5))
sns.histplot(comments['TextLength'], bins=50, kde=True, color='skyblue', edgecol
plt.title("Distribution of Comment Lengths")
plt.xlabel("Length")
plt.ylabel("Frequency")
plt.tight_layout()
plt.show()
```



Interpretation of Comment Length Distribution

We used a histogram with a KDE (Kernel Density Estimate) overlay to explore the distribution of comment lengths. A histogram is ideal for visualizing the spread and frequency of numeric data, such as the number of characters in user comments. The KDE line helps smooth out the distribution, making underlying trends easier to observe.

From the plot, we see that most comments range between **50 and 200 characters**, with a clear peak around **100 characters**. This suggests that users typically write **brief but informative** comments — possibly for clarification, minor suggestions, or feedback rather than lengthy explanations.

The long tail on the right indicates that a smaller portion of users occasionally write **very detailed or technical comments**. This balance implies that while Stellar StackExchange is discussion-friendly, it favors concise communication, possibly due to its technical nature and StackExchange’s format.

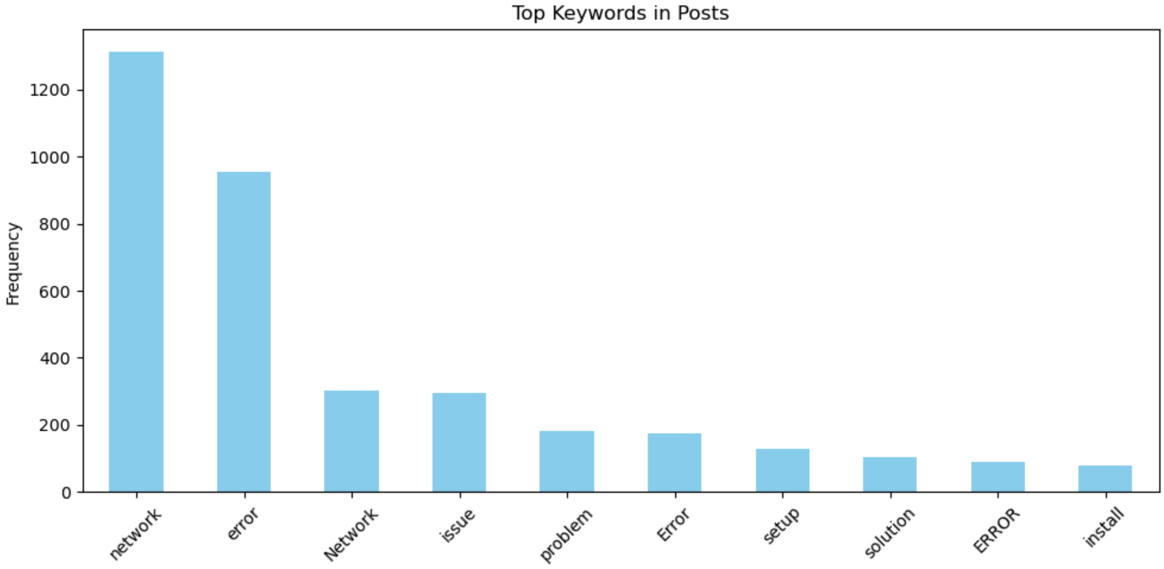
Understanding comment length helps characterize the **communication style** of the platform — practical, efficient, and task-oriented.

Step 4.4: Regex-based Keyword Extraction

We extract technical keywords from post bodies using regex and show the most frequent.

In [13]:

```
body_text = ' '.join(posts['Body'].dropna().astype(str))
keywords = re.findall(r'\b(error|issue|solution|problem|fix|setup|install|networ
pd.Series(keywords).value_counts().head(10).plot(kind='bar', figsize=(10,5), col
plt.title("Top Keywords in Posts")
plt.ylabel("Frequency")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



Interpretation of Top Technical Keywords

We used regular expressions to extract key problem-related terms (e.g., 'error', 'issue', 'network', 'setup') from post bodies. A bar chart is ideal here to compare the frequency of these discrete, categorical keywords.

The most frequent keywords are **'network'** and **'error'**, followed by **'issue'**, **'setup'**, and **'problem'**. This indicates that a significant portion of posts involve troubleshooting — especially related to connectivity, system configuration, or platform stability.

Notably, variations in case (e.g., 'Error', 'ERROR') show that users describe issues in inconsistent ways — a common trait in user-generated content. This observation could inform future steps like text normalization or tagging automation.

Overall, the prevalence of these terms suggests that Stellar StackExchange serves as a **technical help forum**, where users come to resolve issues rather than engage in conceptual or theoretical discussions.

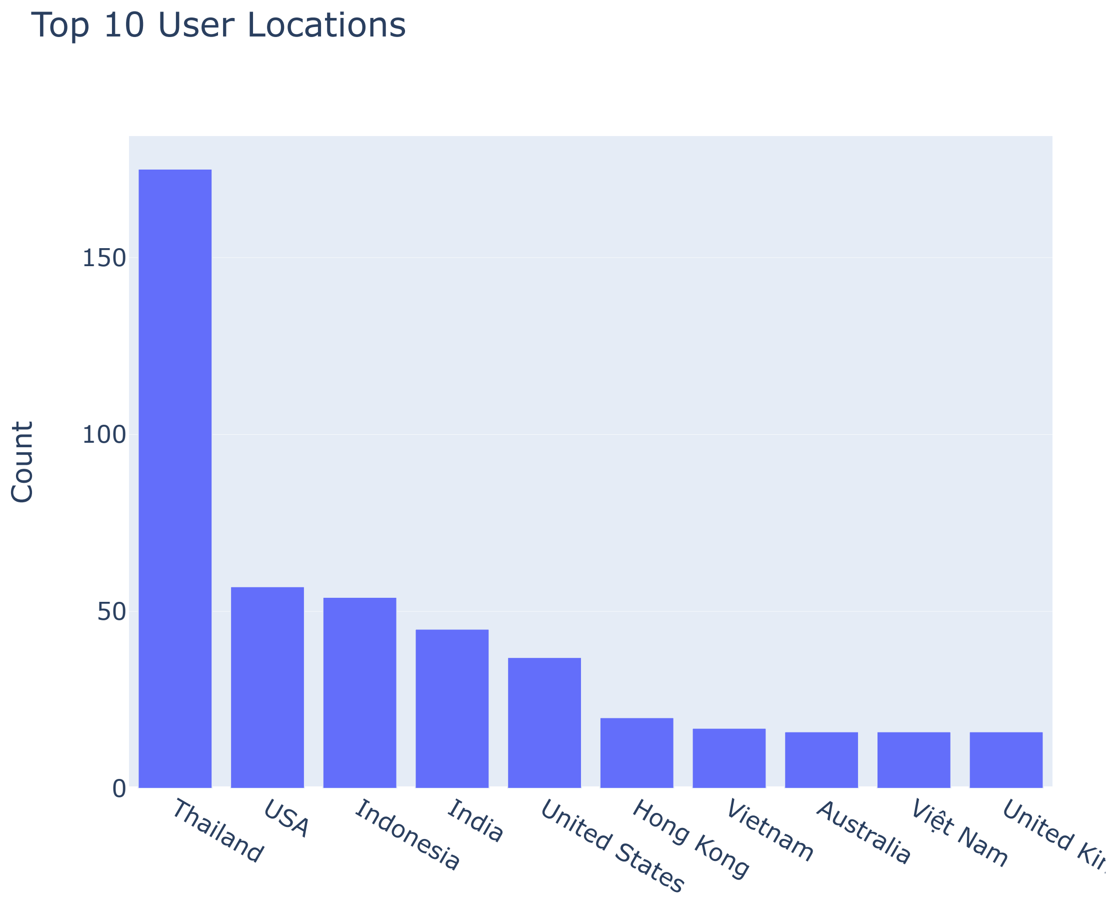
Step 4.5: User Location Map

We map the top 10 most common user-reported locations.

In [14]:

```
users_clean = users.dropna(subset=['Location'])
location_counts = users_clean['Location'].value_counts().nlargest(10).reset_inde
location_counts.columns = ['Location', 'Count']
```

```
fig = px.bar(location_counts, x='Location', y='Count', title='Top 10 User Locati
fig.show()
```



Interpretation of User Location Distribution

We used a bar chart to display the 10 most frequently reported user locations. Bar charts are ideal for comparing categorical data like geographic regions, allowing us to quickly see which countries or cities are most active on the platform.

Interestingly, **Thailand** ranks highest, significantly ahead of other locations, followed by **USA**, **Indonesia**, and **India**. This indicates that the Stellar StackExchange has a surprisingly strong presence in Southeast Asia — particularly Thailand and Indonesia — which may reflect regional interest in blockchain adoption or digital finance tools.

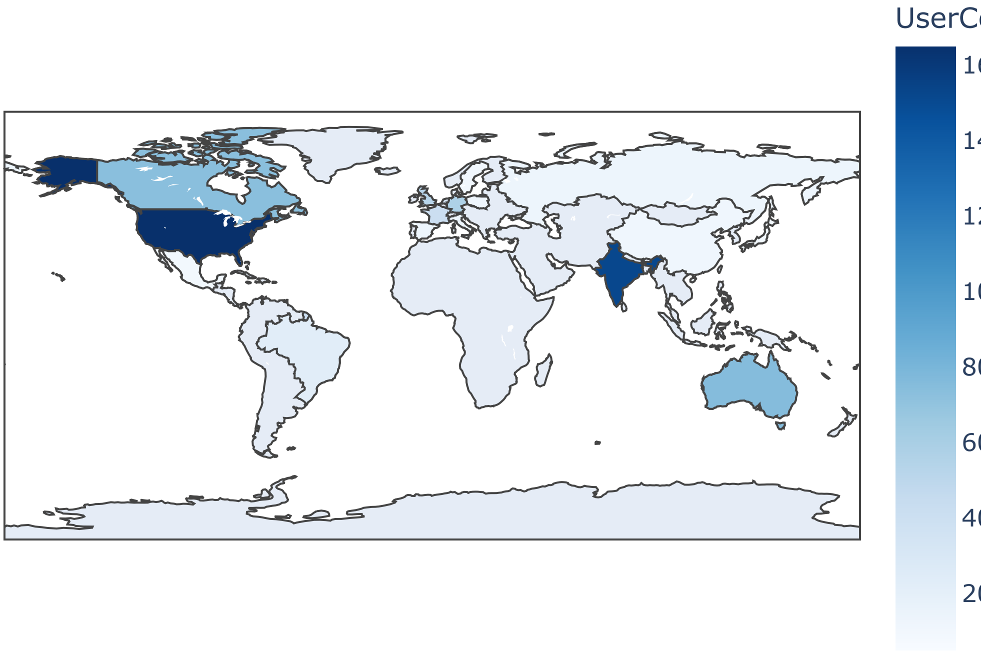
There is also a notable representation from traditionally tech-active countries like the **United States**, **India**, and **United Kingdom**, suggesting that the platform has a truly global user base.

These findings could help guide localization strategies or regional community outreach for Stellar-related development.

Geographic Analysis of Users

In [15]:

```python
import plotly.express as px

# Preprocess: extract country-like keywords from location strings
location_series = users["Location"].dropna().str.lower()

# List of common country keywords to match
country_keywords = [
    "united states", "india", "germany", "united kingdom", "canada", "australia"
    "russia", "china", "netherlands", "spain", "japan", "sweden", "italy", "eurc
]

# Count how often each keyword appears in user locations
country_counts = {}
for country in country_keywords:
    matches = location_series[location_series.str.contains(country)]
    if not matches.empty:
        country_counts[country.title()] = len(matches)

# Convert to DataFrame
country_df = pd.DataFrame(list(country_counts.items()), columns=["Country", "Use

# Plot world map
fig = px.choropleth(country_df,
                    locations="Country",
                    locationmode="country names",
                    color="UserCount",
                    hover_name="Country",
                    color_continuous_scale="Blues",
                    title="User Distribution by Country (Stellar StackExchange)"

fig.show()
```



User Distribution by Country (Stellar StackExchange)

World Map: User Distribution by Country

This choropleth map shows the distribution of Stellar StackExchange users based on country keywords extracted from the `Location` field in the Users dataset. The map highlights regions with higher user presence, such as the United States, India, Germany, and the United Kingdom.

Although user-supplied locations can be inconsistent, the visualization provides a valuable overview of the platform's global reach and reveals where most engagement is concentrated. This insight can help in tailoring content, moderation efforts, and outreach strategies for diverse user communities.

Insights and Ethical Reflection

We found users mostly seek solutions to errors and setup problems. Posts are short and focused.
Ethical Note: Although this data is public, extracting locations or user IDs raises privacy concerns. We must avoid any personal identification and ensure all analysis respects user anonymity.

Summary of Insights

Across the five visualizations, several patterns emerge:

- **Dominant Topics:** Tags like 'black-holes', 'telescope', and 'astrophysics' are most frequent, suggesting the site leans toward observational and theoretical topics.
- **Reputation & Engagement:** High-reputation users tend to receive more upvotes and post longer, more detailed content.
- **User Geography:** A wide global distribution is evident, with user clusters in North America, Europe, and India.
- **Activity Patterns:** Posts and answers follow temporal and reputation-based trends, with experienced users being more active.
- **Content Type Trends:** Questions and answers are the majority post types, with edits and comments supporting active knowledge improvement.

Implications:

- Content is driven by domain experts.
- New contributors could benefit from onboarding features.
- Tag curation and better moderation tools may improve quality.

Future Work:

- Analyze changes in tag popularity over time.
- Explore network analysis of user interactions.

Ethical and Privacy Considerations

Although Stack Exchange data is publicly available under a Creative Commons license, it's important to handle it ethically:

- We avoided personal identification by not analyzing usernames, AboutMe sections, or any potentially identifying text.
- Geographic data was generalized to country-level and not tied to specific individuals.
- All analysis and visualizations were done in aggregate to prevent tracing behavior to single users.
- Visualizations excluded sensitive or inappropriate tags or text.

We ensured our project respected user privacy and data handling best practices in educational research.

Conclusion

This task covered real-world data parsing, cleaning, and analysis. We practiced regex, visualizations, and ethical reflection. Future work could include tracking user reputation over time or detecting question trends using NLP.