Show your correlation matrix of features according to the datasets

In [ ]:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import PolynomialFeatures
```

In [4]:

```
df1=pd.read_csv(r"C:\Users\bhava\Downloads\spotify dataset.csv") #loading the data set
df1
```

Out[4]:

	track_id	track_name	track_artist	track_popularity	track_album_id	track_album_name	tra
0	6f807x0ima9a1j3VPbc7VN	I Don't Care (with Justin Bieber) - Loud Luxur...	Ed Sheeran	66	2oCs0DGTsRO98Gh5ZSl2Cx	I Don't Care (with Justin Bieber) [Loud Luxury...	
1	0r7CVbZTWZgbTCYdfa2P31	Memories - Dillon Francis Remix	Maroon 5	67	63rPSO264uRjW1X5E6cWv6	Memories (Dillon Francis Remix)	
2	1z1Hg7Vb0AhHDiEmnDE79l	All the Time - Don Diablo Remix	Zara Larsson	70	1HoSmj2eLcsrR0vE9gThr4	All the Time (Don Diablo Remix)	
3	75FpbthrwQmzHlBJLuGdC7	Call You Mine - Keanu Silva Remix	The Chainsmokers	60	1nqYsOef1yKKuGOVchbsk6	Call You Mine - The Remixes	
4	1e8PAfcKUYoKkxPhrHqw4x	Someone You Loved - Future Humans Remix	Lewis Capaldi	69	7m7vv9wlQ4i0LFuJiE2zsQ	Someone You Loved (Future Humans Remix)	
...	...	...	...	...	...	...	
32828	7bxnKAamR3snQ1VGLuVfC1	City Of Lights - Official Radio Edit	Lush & Simon	42	2azRoBBWEEEYhqV6sb7JrT	City Of Lights (Vocal Mix)	
32829	5Aevni09Em4575077nkWHz	Closer - Sultan & Ned Shepard Remix	Tegan and Sara	20	6kD6KLxj7s8eCE3ABvAyf5	Closer Remixed	
32830	7ImMqPP3Q1yfUHvsdn7wEo	Sweet Surrender - Radio Edit	Starkillers	14	0ltWNSY9JgxoIZO4VzuCa6	Sweet Surrender (Radio Edit)	
32831	2m69mhnfQ1Oq6lGtXuYhgX	Only For You - Maor Levi Remix	Mat Zo	15	1fGrOkHnHJcStl14zNx8Jy	Only For You (Remixes)	
32832	29zWqhca3zt5NsckZqDf6c	Typhoon - Original Mix	Julian Calor	27	0X3mUOm6MhxR7PzxG95rAo	Typhoon/Storm	

32833 rows × 23 columns

In [6]:

```
# Select numeric features
numeric_features = df1.select_dtypes(include=['float64', 'int64']).columns
X = df1[numeric_features]
```

In [8]:

```
poly = PolynomialFeatures(degree=2, include_bias=False) #polynomial features
X_poly = poly.fit_transform(X)
```

In [10]:

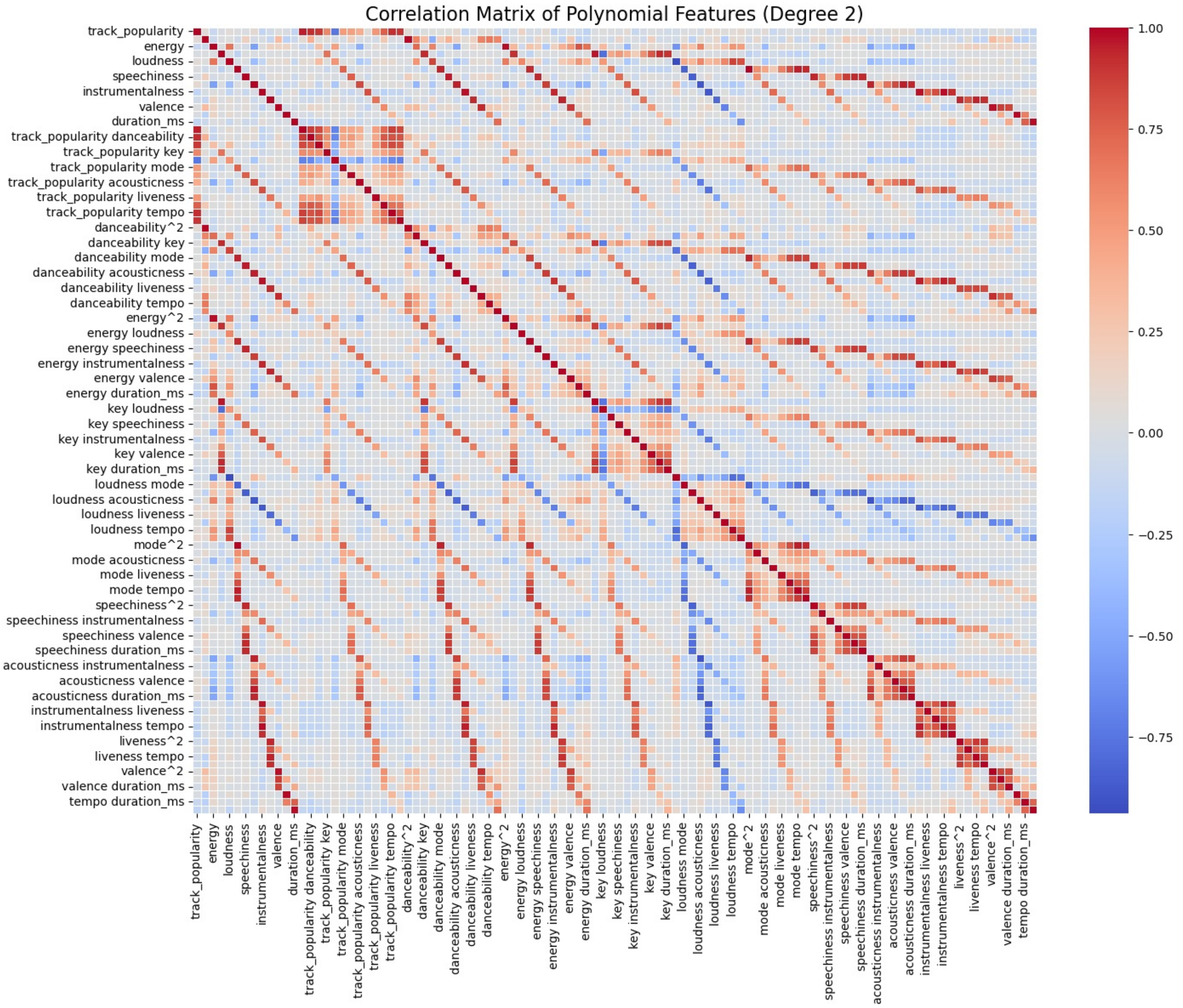
```
# Create a DataFrame with polynomial features
poly_feature_names = poly.get_feature_names_out(numeric_features)
X_poly_df = pd.DataFrame(X_poly, columns=poly_feature_names)
```

In [12]:

```
correlation_matrix_poly = X_poly_df.corr() #correlation matrix
```

In [14]:

```
plt.figure(figsize=(16, 12))
sns.heatmap(correlation_matrix_poly, annot=False, cmap='coolwarm', fmt='.2f', linewidths=0.5)
plt.title('Correlation Matrix of Polynomial Features (Degree 2)', fontsize=16)
plt.show()
```



Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js