In [4]:

```
import pickle
import warnings

import matplotlib as mpl
import pandas as pd
from matplotlib import pyplot as plt

from utils.plots import bar_metrics


# We will use deprecated models of statmodels which throw a lot of warnings to use more
warnings.filterwarnings("ignore")
plt.style.use('bmh')
mpl.rcParams['axes.labelsize'] = 14
mpl.rcParams['xtick.labelsize'] = 12
mpl.rcParams['ytick.labelsize'] = 12
mpl.rcParams['text.color'] = 'k'
mpl.rcParams['figure.figsize'] = 18, 8

with open('results/scores.pickle', 'rb') as handle:
    resultsDict = pickle.load(handle)

# Load our results from the model notebook
with open('results/predictions.pickle', 'rb') as handle:
    predictionsDict = pickle.load(handle)
```

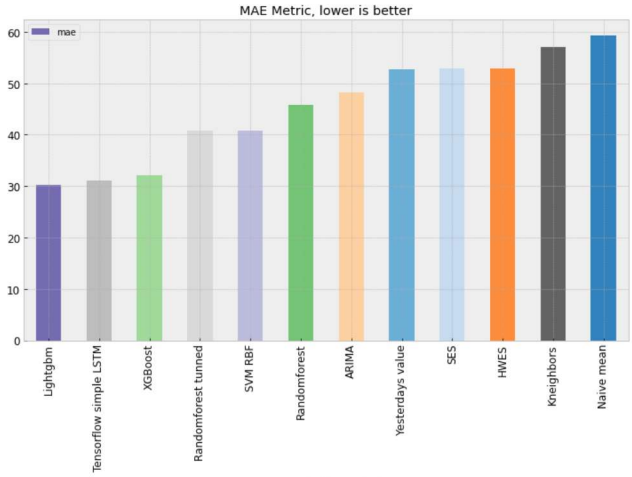
Evaluation Metrics

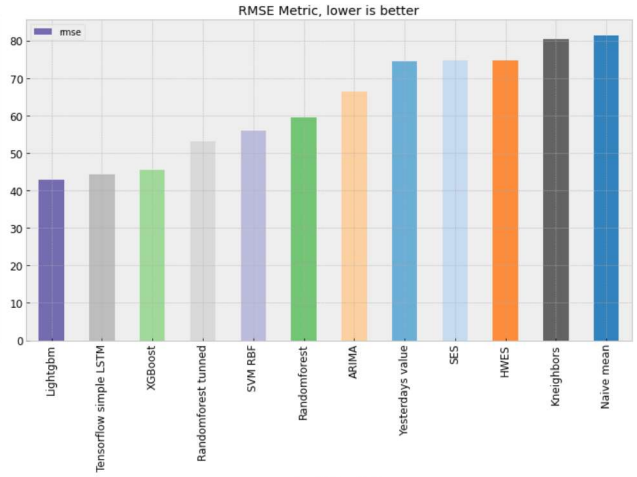
There are many measures that can be used to analyze the performance of our prediction so we will be using the top 4 most used metrics for time series forecasting. Each of this metrics are different from the others in the way the test our predictions.

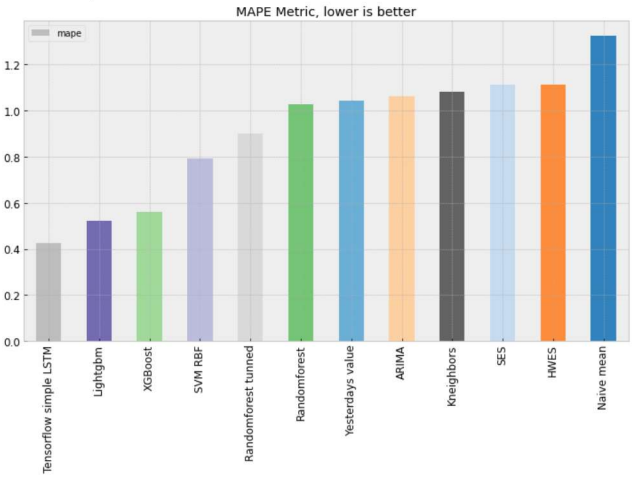
- Mean Absolute Error (MAE)
- Mean Absolute Percentage Error (MAPE)
- Root Mean Squared Error (RMSE)
- R2 Coefficient of determination (r2)

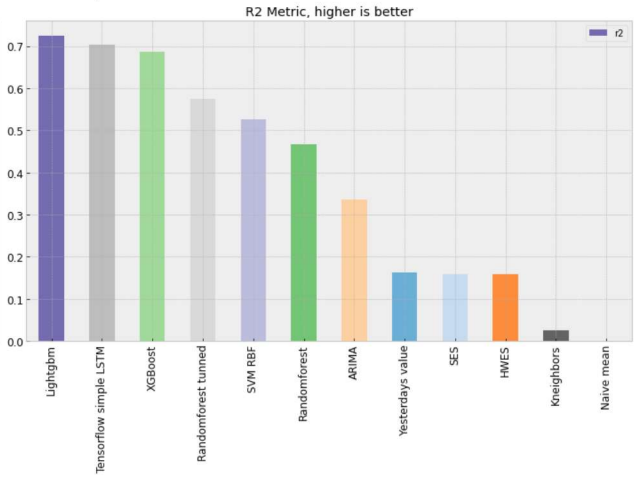
In [6]:

```
bar_metrics(resultsDict)
```


MAE Metric, lower is better
mae
0
10
20
30
40
50
60
Lightgbm
Tensorflow simple LSTM
XGBoost
Randomforest tunned
SVM RBF
Randomforest
ARIMA
Yesterdays value
SES
HWES
Kneighbors
Naive mean


RMSE Metric, lower is better
rmse
0
10
20
30
40
50
60
70
80
Lightgbm
Tensorflow simple LSTM
XGBoost
Randomforest tunned
SVM RBF
Randomforest
ARIMA
Yesterdays value
SES
HWES
Kneighbors
Naive mean


MAPE Metric, lower is better
mape
0.0
0.2
0.4
0.6
0.8
1.0
1.2
Tensorflow simple LSTM
Lightgbm
XGBoost
SVM RBF
Randomforest tunned
Randomforest
Yesterdays value
ARIMA
Kneighbors
SES
HWES
Naive mean


R2 Metric, higher is better
r2
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Lightgbm
Tensorflow simple LSTM
XGBoost
Randomforest tunned
SVM RBF
Randomforest
ARIMA
Yesterdays value
SES
HWES
Kneighbors
Naive mean