# Solution Design document

- *Developed by Aleksandr Vikentev, Ditesh Verma and Yanqing Yu*

**1. Project title**
Explore and demonstrate ‘Advanced Intraday Trading Strategies’ having a high sharpe ratio.

**2. Objective**
We aim to explore the effectiveness of various technical indicators leveraging minute level Open, High, Low, Close, Volume and open interest data in the Indian equity futures market. We specifically demonstrate mean reversion, momentum and seasonality strategies that have short holding periods ranging from one minute to five minutes and have sharpe ratios ranging from 0.5 to over 5, excluding any transaction costs.

**3. Project Specifications & Solution Architecture**

**3.1 Data Collection**: We are provided with minutely Open, High, Low, Close, Volume and Open interest data in csv files for the last two years. Each file is a month’s data for 186 tickers out of which four are index and rest are single name equity futures. We upload this data into our github public repository so it is accessible to our google colab notebook. The code imports data using the TickerData class, which also resides in the GitHub repository.

**3.2 Data Processing**: The class *TickerData* is used to import data from the repository and process it. Data for the 'Bank Nifty' futures contract (BANKNIFTY) is the main dataset used for the technical feature analysis. We also process 182 single name equity tickers data in the cross sectional reversal and momentum strategy. The code splits the data into in-sample and out-of-sample sets for model development and evaluation. This is done to avoid overfitting and to evaluate how well the model generalizes to unseen data. First 80% of the data is used as in sample and last 20% is used as out of sample.

**3.3 Analysis and Reporting**: The *Exploratory* class is used to conduct exploratory research on the data. The *Strategy* class helps in creating features based on technical indicators like Bollinger Bands, ATR and MACD are generated and saved as csv files for backtesting. Function test_robustness is developed for robustness testing - it takes parameters such as lookback window, band tolerance and lags as arguments. Another function plot_robustness helps to visualize the results in an effective manner by plotting a chart grid.

**3.4 User Interface**: We utilized Google colab to run the jupyter notebooks. These notebooks is used as an interface to import the data, processing and plotting results for various strategies. As we analyzed lots of technical indicators and strategies, we had to split the work into multiple jupyter notebooks. The file names and specification are mentioned in the readme document.

**Summary**
The coding work is modularized based on their logical functions such as data processing, exploratory analysis, feature engineering and Backtesting. This modularized approach really

helps in cutting down time to create new strategies and efficiently demonstrating the effectiveness of the developed strategies. Reusability of the modular code meant that every member of the group was up to date on any changes and no one had to reinvent the wheel. The usage of github repository further enhances the collaboration by allowing everyone to edit the code and contribute to the project in parallel. This really enhanced our productivity which eventually helped us analyze over 20 technical indicators and finally over 10 strategies were developed with a high sharpe ratio.