## Use case 1: Calculate Option Prices and display Volatility graph

## Primary Actor:

Trader

## Stakeholders and Interests:

Trader- wants to accurately calculate option prices across multiple stock options

Investor (User/Stock holder): wants to make a choice to call/put depending upon the option prices

Stock Exchange Service: provides stock data to the trader

## Pre-conditions:

* Trader has access to the Option Pricer system
* Trader has retrieved data from the investor

## Post-conditions:

* System generates an accurate and consistent option price
* System generates volatility graph

## Main Success Scenario:

* 1. Trader selects the region of Stock Option
  2. System provides list of algorithms to choose from
  3. Trader selects one of the algorithms from the list
  4. Trader enters input parameters like Current Stock price, Duration, Expected growth rate, Stock price volatility, Risk-free interest rate, Strike price, Option to call or put
  5. System calculates the option price based on the input values
  6. System gives an option to generate the volatility graph
  7. System provides an option to run one of the alternative algorithms for the same input values or start a new request (Go to 1.1).

## Alternative Flows:

1. Data Inconsistency in input variables
   1. Trader enters input variable in incorrect format
   2. System highlights the conflicting field and prompts trader to re-enter values
   3. Trader enters new/correct value
2. At any time, aborting the ongoing process of calculating option price:
   1. Trader selects cancel button to abort the process of calculation from any stage after 1.1
   2. System goes back to new request (1.1)
3. At any time, System fails:

To support recovery and correct calculation, ensure the last session is restored from the same step of the scenario.

* 1. System crashes due to some anomaly
  2. System reboots and reconstructs the last state

## Special Requirements:

* System should generate accurate results with variance in less than 0.5 seconds
* System should be able to convert stock option prices to multiple currencies as per the closing exchange rate on the last business day
* Trader will be able to add a new algorithm into the system

## Technology and Data Variations List:

* Currencies maybe any of USD, YEN, POUND, EURO etc.
* Trader enters input parameters through keyboard
* Trader can use device used on any of the following operating system – UNIX, iOS and Windows.

## Priority:

High

## Frequency of occurrence:

Continuous use during business hour. Trader may use it frequently throughout the day which may peak around 10 – 15 times per hour

## Open Issues:

* How will stock price information be retrieved from the stock exchange?

## Use Case 2: Adding additional algorithm

## Main Success Scenario:

Trader selects option to add her/his algorithm into the system. System prompts trader to enter algorithm for one or many regions. Trader enters algorithm. System compiles the algorithm and checks for syntactical errors. System gives an option to re-enter algorithm or abort or confirm the addition. Trader confirms addition of new algorithm. System adds algorithm as an option to the list of calculating stock option prices.

## Alternate Flows:

* If system finds errors in the input, it signals error and prompts trader to correct the errors or abort the process
* If trader aborts the process, system discards algorithm
* If system detects an error in importing the algorithm, it signals importing error to the trader and sends issue to the support team for resolution