You will need to create a C# .Net application that provide a list of instruments and their respective details. All of this information should be available via an API. Your solution should contain the following:

1. An API with 4 endpoints to service all user requests
2. A daemon (daemon #1) that will poll the Yahoo Finance API to get all the required data
3. A daemon (daemon #2) that will send emails
4. Unit tests with reasonable code coverage
5. A database where all data is stored

Technical specifications:

The API:

1. Should have 4 methods to get the following details:
   a) List of all instruments (GET) - This should show the instrument name, symbol, type
   b) Instrument summary (GET) - This should show instrument short and long name, symbol, currency, currency symbol, exchange name, price, summary and recommendation
   c) Instrument price (GET) – This should retrieve the instrument's current price
   d) Price Alert subscription (POST) – This should keep track of when a user wants an alert that a price for a particular instrument is met in order to send an email to the user

Daemon #1:

1. Should download the instrument details from https://rapidapi.com/apidojo/api/yh-finance
   a) Yahoo endpoint - market/v2/get-summary

Property Name	Response JSON path
Symbol	MarketSummaryAndSparkResponse.Result[0].Symbol
Short Name	MarketSummaryAndSparkResponse.Result[0].ShortName
Type	MarketSummaryAndSparkResponse.Result[0].QuoteType

   b) Yahoo endpoint - stock/v2/get-summary

Property Name	Response JSON path
Short Name	Price.ShortName
Long Name	Price.LongName
Symbol	Symbol
Currency	Price.Currency
Currency Symbol	Price.CurrencySymbol
Exchange Name	Price.ExchangeName
Price	Price.RegularMarketPrice.Raw

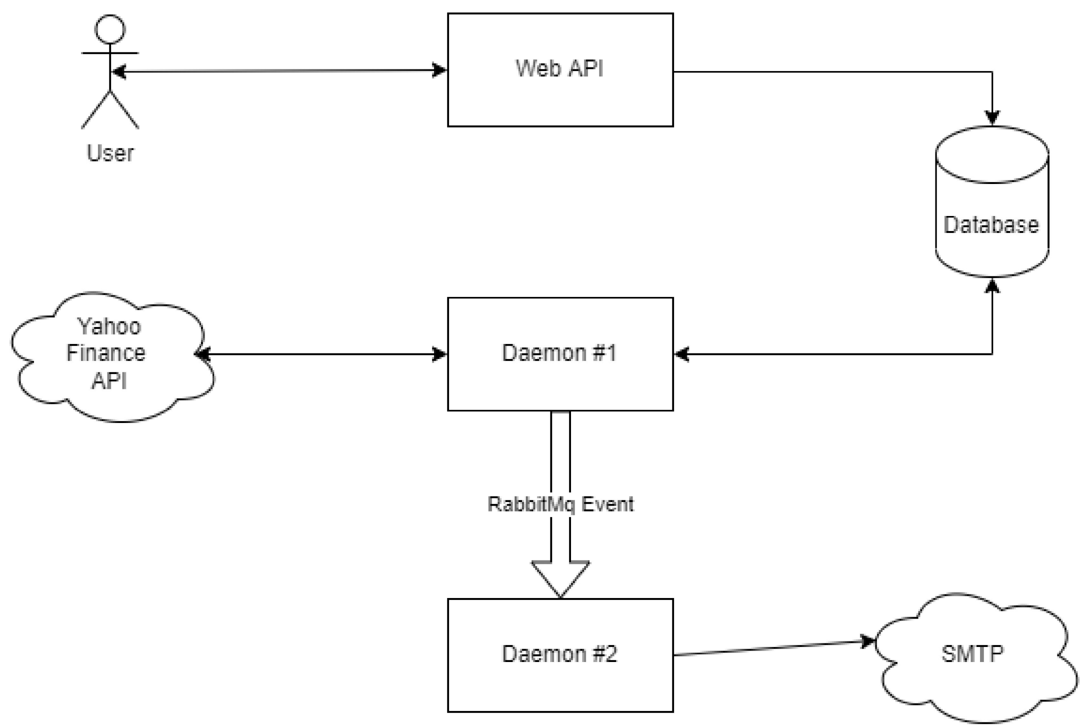
Summary	SummaryProfile.LongBusinessSummary
Recommendation	FinancialData.RecommendationKey

2. All this information should be stored in a database
3. This daemon will also poll the Yahoo API for price updates every 5 seconds. Price updates should be stored in a cache for faster retrieval
4. If an instrument's price hits a user's price alert, a rabbitmq event should be published to Daemon #2

Daemon #2:

1. This daemon should listen to rabbitmq events coming from daemon #1
2. This daemon should send an email to the user whose price alert was met.

High level overview:



Glossary:

Instrument - Financial instruments are assets that can be traded, or they can also be seen as packages of capital that may be traded. Most types of financial instruments provide efficient flow and transfer of capital all throughout the world's investors. These assets can be cash, a contractual right to deliver or receive cash or another type of financial instrument, or evidence of one's ownership of an entity. Eg. Apple, Microsoft, S&P 500