**Module Description, Module Implementation (Phase 1) Using Agile.**

**Procedure:**

**1- Module Description-**

Main module in the project " Bitcoin Price Prediction Using AI & ML" for Implementation is-

**Data collection & Preparation**

Each phase must be completed before the next phase can begin and there is no overlapping in the phases.

* Data Collection plays important role in this project .Without data we can't predict the price of Bitcoin. If we want to predict bitcoin price ,we need atleast last 2-3 days data to predict and if we want to predict the price for last 1 years in form of graphs and diagrams we need large amount of data and for that we need to collect data online.
* Data preprocessing is a data mining technique that involves transforming raw data into an understandable format.
* Real-world data is often incomplete, inconsistent, and/or lacking in certain behaviors or trends, and is likely to contain many errors.
* Name and little bit description of programming language used to implement the module -

If we want to predict the price of bitcoin ,the only Programming language is PYTHON.

like - #import the libraries

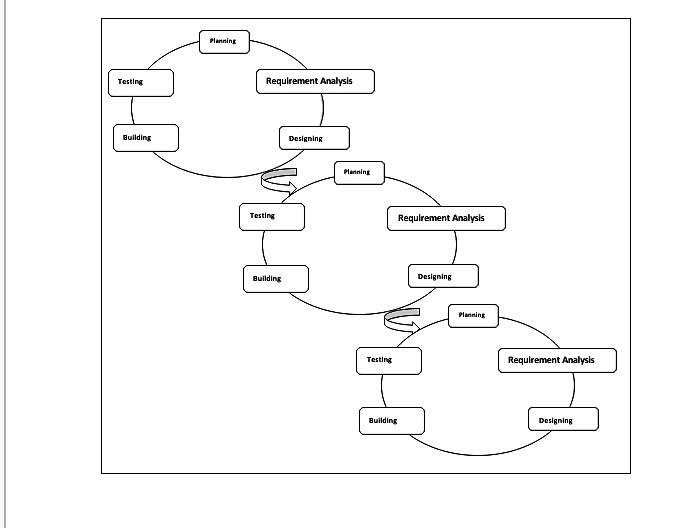
import mumpy as np

import pandas as pd

................................

**2- Module Implementation (Phase 1) Using Agile-**

draw the diagram of agile method as given below.



**Iteration 1-**

a) Planning: - To implement this ,first decided which data sets and time-step to use.The financial data was only available on a daily basis for commercial reasons. Two different data sources are collected ;

> The first consisting of historical BTC/USD exchange rate data(bitcoin data in real time) .

> The other of tweets (tweets data in real time).

b) Requirement Analysis: - A dedicated server, allowing for uninterrupted continues data gathering to collect datasets.

c) Designing: - Historical price points for Bitcoin were gathered daily from CoinDesk

publicly available Quandl's API.

d) Building: - Depending on the time interval length of the requested data, CoinDesk returns different levels of detail, such as transaction based or aggregated in the form of OHCL (open,high,close,low).CoinDesk Quandl's API request for USD/BTC exchange price data.

\ u rl { h t tp :// api . coindesk . com/ c h a r t s /da ta ? ou tpu t=csv\&da ta

,→ = cl o s e\&index=USD&s t a r t d a t e =2017−05−09&enddate

,→ =2017−05−09\&exchanges=bpi&dev =1 }

An interval length of a day returns price data for every minute.

**Iteration 2-**

a) Planning: - Gathering tweets data in real-time are tweets processing & tweets sentiment analysis.

To collect data for the sentiment analysis Twitter’s streaming Quandl's API was used in combination with Tweepy.

b) Requirement Analysis: - Tweepy, an open source framework written in Python, facilitates tweet collection from Twitter’s Quandl's API .

c) Designing:-Tweepy allows for filtering based on hashtags or words, and as such was considered as an efficient way of collecting relevant data.

d) Building: - The filter keywords were chosen by selecting the most definitive Bitcoincontext words, for example "cryptocurrency" could include sentiments towards other cryptocurrencies, and so the scope must be tightened further to only include Bitcoin synonyms. These synonyms include: Bitcoin, BTC, XBT and satoshi.

Example function that gathers a stream of filtered tweets breaklines

def b tc \_ twee t \_ s t re am ( ) :

api = twi t te rAP IConnec tion ( )

l i s t e n e r = S tdOu tLi s tene r ( )

s tream = tweepy . Stream ( api . auth , l i s t e n e r )

s tream . f i l t e r ( t r a c k =

[ ’ b t c ’ , ’ b i t c o i n ’ , ’ xb t ’ , ’ s a t o s hi ’ ]

, languages = [ ’ en ’ ] )

**Iteration 3-**

a) Planning: - The data is available in three different formats i.e JSON, XML and CSV. Data is downloaded in the .csv format. Size of data is more than 200KB. It has a more than 2000 data records (each record corresponds to a day) consisting of Bitcoin open, high, low, closing price and volume of Bitcoin(USD).

b)Requirement Analysis: - For this phase, we need CSV file which is having bitcoin historical & tweet data & the convert it into datasets & create dataframes (dataframes in python are used to store data tables) & dataframes are actually a list of vectrs which will store all datasets.

c)Designing: - Dataset is referred toas the Norm dataset. The latter one includes the mean normalization of all features except that of price. This is named as UNorm dataset. the magnitude of the future price is determined by the previous prices and the deviations from the previous price is captured by other features present in the dataset as described above.

d) Building: -After downloading the datasets , by scaling the data by min-max normalization & preprocessing, we need to split the data and divide them into 2 parts- train data(80%) & test data(20%) & merging the train data & test data of whole datasets.