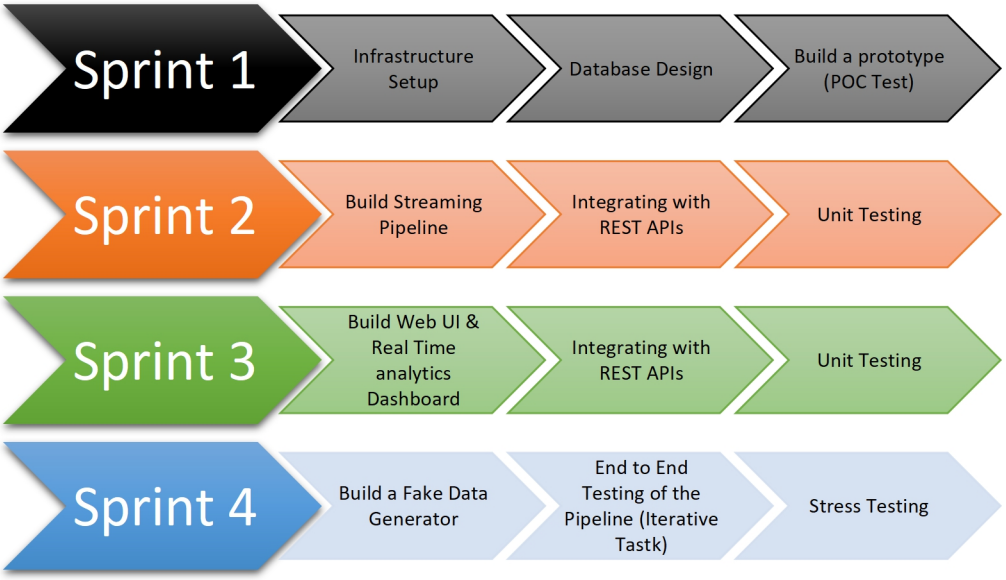
Stream Processing Analytics
For
Community App

Solution & Approach

In this solution we are building a Community App named as ***"Streamer"*** where users can create multiple communities and posts there Quotes, Images or Videos relevant to there community. Each post can be like it, comment on and even share the existing post.
While creating this application, we would need certain methodology to follow and the right technology stack which we can use to successfully implement the functionality needed. Following are the steps presenting how this can be achieved step by step. As we can see in the above mentioned problem we have to keep in mind that this solution needs to be scalable, fault tolerant, and provide real time insights about the data.

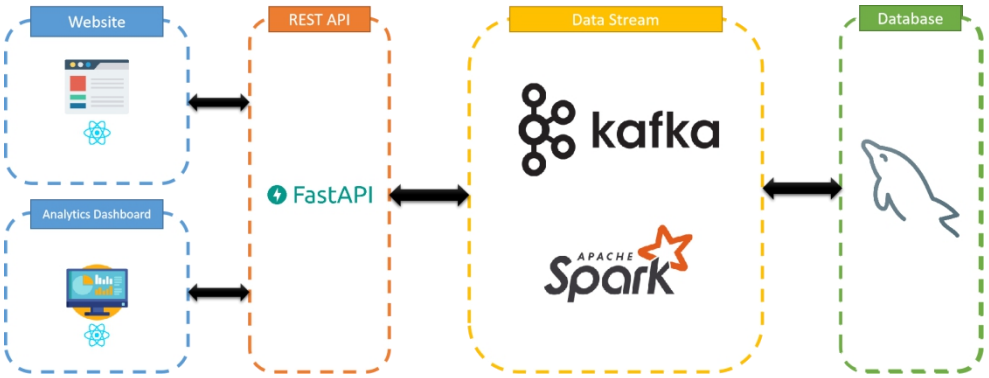
SDLC method : As we know the application is related to social media, it must pass through a lot of testing and staging to get the perfect and most reliable software to the users. The best SDLC method to use is the Agile Method which launches an application in successive iterations and the design and code get cleaner and cohesive as the iterations proceed forward. Using this method, one can get a working prototype or MVP directly in a short time and then check for bugs and feasibility.



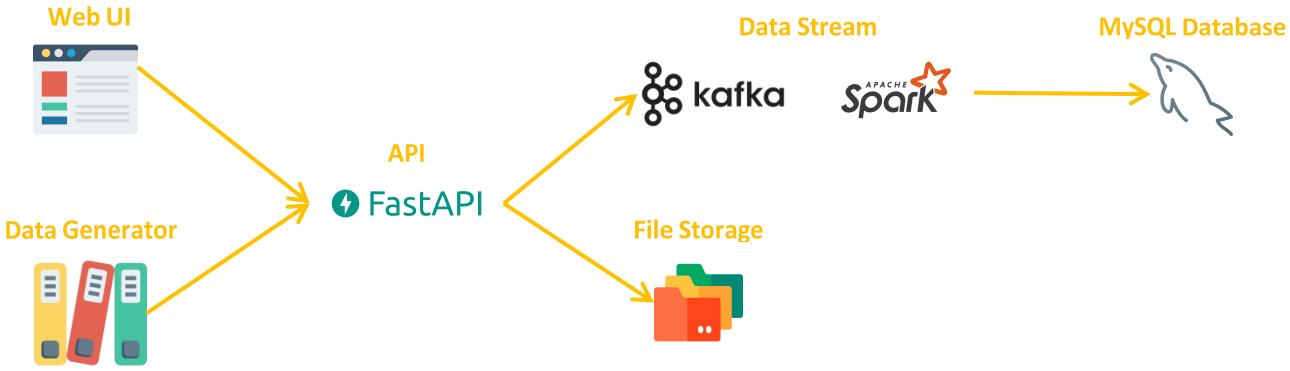
Requirements Gathering and Feasibility : As we have been already given the requirements and assumptions. To be followed while building this application.

Choosing the right tech stack : Social Media platforms receives a lot of load and data on the servers, so it is most important to choose the suitable tech stack. Let's point the best technology that one can use to create the perfect application.

High Level Architecture :

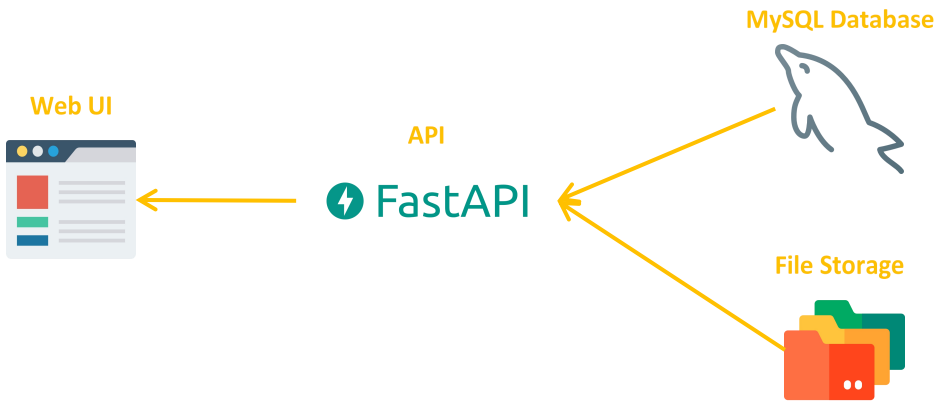


Streaming Data-flow Architecture : We have broken the workflow into 6 components - *Web UI*, *Data Generator*, *API*, *Data Stream*, *File Storage*, *MySQL Database*. *Web UI* is user application from where data will be ingested through *REST APIs* , and in parallel we are generating the data through fake *data generator* and that will also be happening through *REST API*. There is back-end that will handle *REST APIs* request and further these data items will be passed to *Data Stream* i.e kafka and pyspark. This *Data Stream* , will handle real time data and will do some pre-processing and Analytics. After the processing of the data it will be ingested to *MySQL Database*. But we need to store the images and videos, for that purpose we are using simple *File Storage* (this might change depending upon the POC).

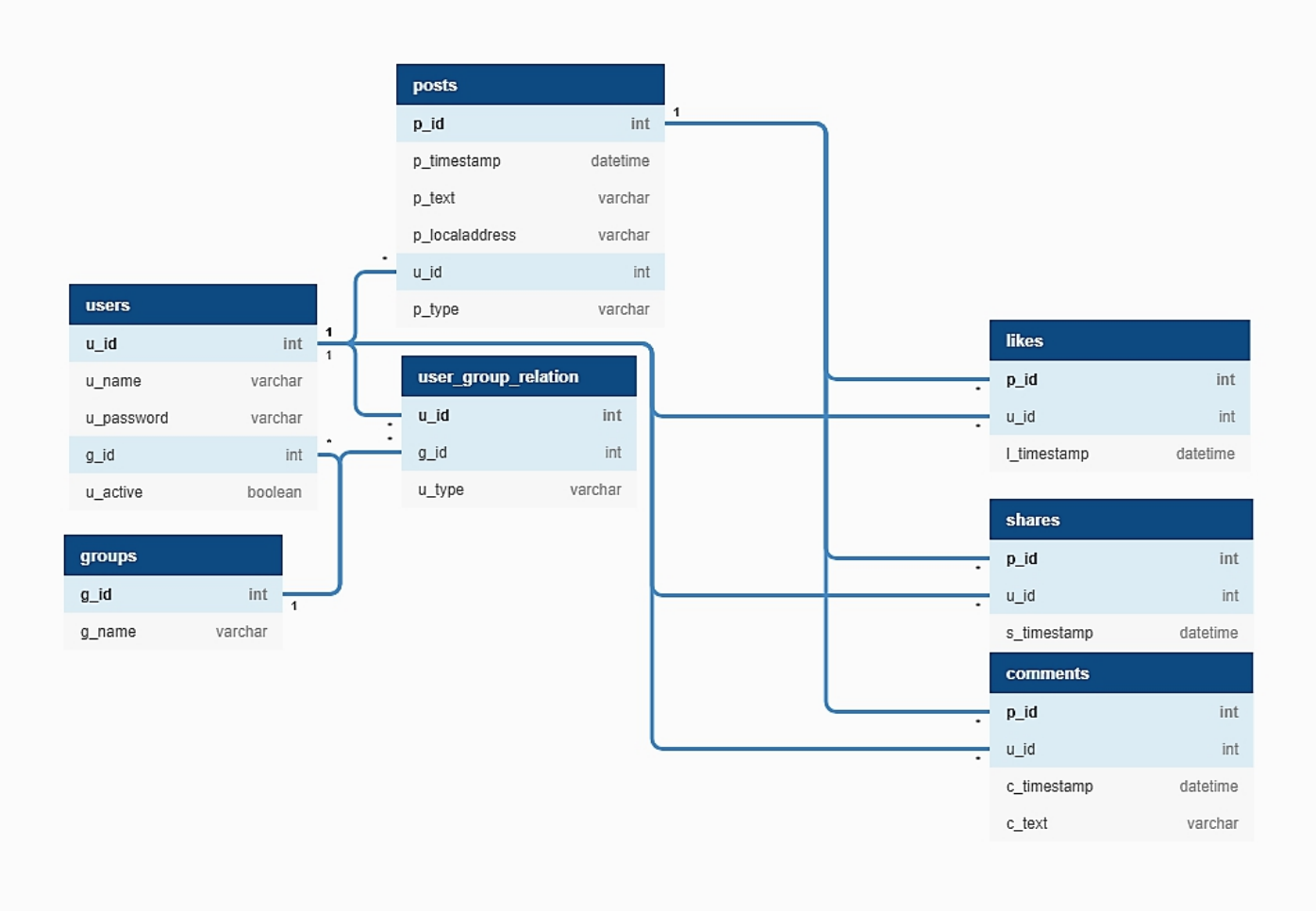


Streaming Data Analytics Framework : Here comes the interesting part where we are planning to implement basic ML algorithm. In this we are using *Data Stream Consumer (Spark)* for Analytics and the response through *REST API.* And from *REST API* we will be visualizing at *Dashboard UI*. This process is a to-and-fro.

End-User Data-flow : This is a simple workflow for viewing data from database, as we don't require any Analytics at user end. We are simply fetching the request through *REST API* from *MySQL Database* and *File Storage* and viewing at *Web UI*.



Database Schema (Planned)



Power Point Presentation : https://drive.google.com/file/d/1a4kEZ9s9ozKX41fUtN65JM1cBTx-ttCX/view?usp=sharing
Video Link : https://drive.google.com/file/d/14TuWweCm1Qt6aExP7b_cTofQg4BFwNaI/view?usp=sharing