Scaleit.Up Project journal

1 Aditya Chouhan, 2 Manikant Prasad, 3 Megha Nair, 4 Vidhya VijayaKumar

Dept. of Software Engineering, San Jose state University
One Washington Square,San Jose, California ,95192, USA

<aditya.chouhan@sjsu.edu>

<manikant.prasad@sjsu.edu>

<megha.nair@sjsu.edu>

<vidhya.vijayakumar@sjsu.edu>

Abstract:

Over the past 20 years, social media such as Twitter has helped us in connecting easily and instantly with anyone across the world . Social media has also provided a platform for young people to learn new skills and technologies and receive real time news and information. Twitter has also helped entrepreneurs and business owners to connect with different customers and sell their products. They can conduct their business and connect to different people across the world even from their home. Social media has helped many artist toshowcase their talents and skills which would otherwise go unnoticed. Twitter broadcasts the tweets for more than 100 million users and provides a platform for free marketing.

Keywords:

Twitter, Oauth, Tweets, Blurbs, Retweets, Twython API, Administrator,Features, React JS & Material UI, Cloud

I.INTRODUCTION

There has been always demand in any business to build a brand value of the company and its product. It is always a good choice to give their employees a platform to advertize for their own company. Social media is a good platform. But there is also a need to guide these employees and help them by giving them a platform on how to build the company image and increase product sales. Our app is designed to fulfill this requirement.

A. Problem Statement

Many corporations/organization do not have any platform on how to restrict their employee in writing anything for their company in social media. Sometimes this becomes a serious issue where employees might not be aware of. There is another concern of employee on what to tweet , how to tweet and how to increase company image value or product sales. If they give time in building such tweets or image tweets, it might happen their regular work which is again a loss to the company. Our application is trying to reduce this woes and concerns of the company.

B. Scope

Our Application handles the below operations for two different kinds of users for any company: marketing admin team and employees.

* Marketing Team: They can upload, edit and delete new tweets,image tweets.

* Employee: They will Oauthentication with their Twitter Account. They can tweet without images, with images and can tweak tweets. They can search tweets of any other employees or anything related to their company and retweets those tweets.

C. User Personas:

1.Sunil Admin User

Scaleit Up App gives Sunil, Marketing Administrator team full access to Mongo Database with a user experience User Interface to change blurbs and images/blurbs. It also has a dashboard which gives the the idea of the no of data in databases when the user login to the Homepage.

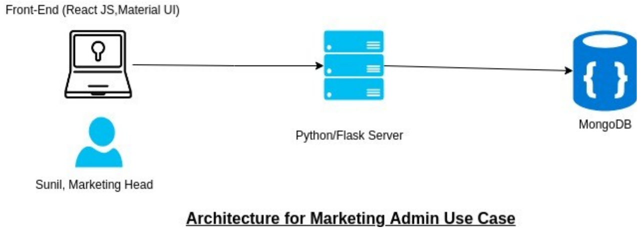
Front-End (React JS,Material UI)
Sunil, Marketing Head
Python/Flask Server
MongoDB
Architecture for Marketing Admin Use Case

Figure 1: Marketing Team Use Case

2. John, Employee:

John ,an employee in the same company can login via his twitter account and tweet directly in twitter. He can directly use the marketing team blurbs or can tweak it to tweet in his twitter account. He can also search and retweets, images and blurbs. The dashboard gives him the no of tweets, re-tweets and images tweet he has done till date once he lands in the home page.

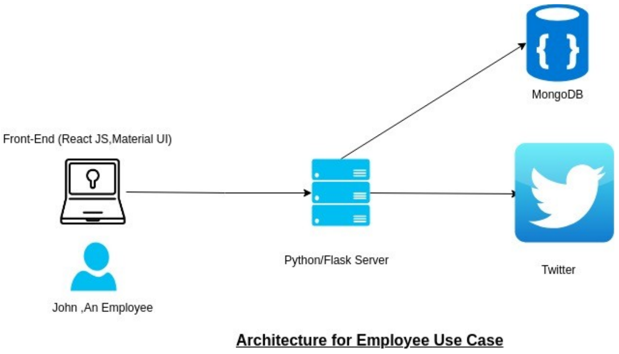
MongoDB
Front-End (React JS,Material UI)
Python/Flask Server
Twitter
John ,An Employee
Architecture for Employee Use Case

Figure 2: Employee Use Case

II. IMPLEMENTATION

A. Databases: We are storing all our tweets, user details and OAUTH credentials in mongoDB (NoSQL) database. We are storing images in Google Firebase. Our REST API (CREATE,GET,DELETE,UPDATE) calls to database is happening via Python/Flask which is our backend system.

B. Backend: Our backend is designed in python as programming language and framework used is flask to connect with our frontend technology. They key to connect to Twitter is Twython API which is available in python language.

C. OAuthentication: We are managing this functionality by redirecting the user to a dynamically generated link by twython API each time an employee logs in to our site. After successful authentication, twitter is giving us back two token, oauth-token and oauth-secret which we are storing it in database along with user details. On logout, this details are deleted.

D. Frontend: Our frontend is designed with React JS and Material UI. We have focussed on good UI design so to incorporate good user experience. Good user experience is very important factor to consider when designing UI.

III FEATURES

Features in our application is divided for two use cases : Admin team and Employee.

A. Marketing Team:

* Dashboard - This view will give details on how many tweets user uploaded.

* Blurbs – This view will show all the blurbs uploaded by team. One can add new tweets, delete or edit it.

* Image Blurbs - This view will show all the image blurbs uploaded by team. One can add new image tweets, delete or edit it.

B. Employee:

* Login – User with login with their Twitter Accunt using Oauth authentication feature.

* Dashboard- This view will give details to user on how many blurbs, image blurbs and retweets user has done till date.

* Blurbs – This view will allow user to tweak tweets and tweet to twitter via Tweet Button.

* Image Blurbs – This view will allow user to tweak tweets only and not images and tweet to twitter via Tweet Button.

* Retweets- In this view, user can seacrh tweets from twitter and retweet those tweets.

IV Cloud Deployment

Our app is deployed on cloud for demo purposes. Cloud deployment has been done on two platforms:

* Heroku – Our frontend is deployed in Heroku using environment variables to connect to our backend system.

* AWS EC2 – Our backend server is deployed in AWS EC2 instance with REST API to which our frontend calls to get a response.

*MLAB – Mlab provides mongoDB as a database as a service. This is again in cloud. We have used just the database feature.

*Google Firebase – This imagestore is to store imageson cloud. Here we have done our Oauth with one of the common project credentials so that user need not give their ids to store images. Images are stored under one user which is under our control.

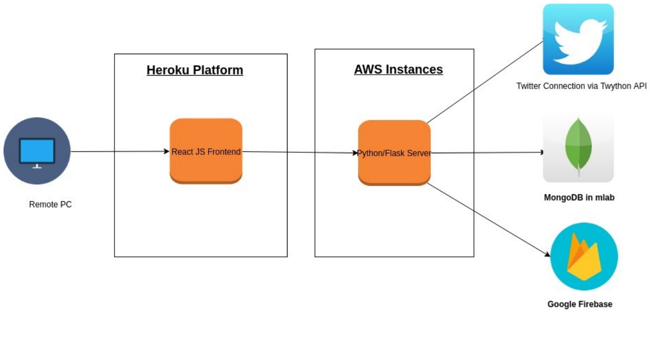

Figure 3 : Cloud Deployment Architecture

V. CONCLUSION

As we have seen above, how this application can be a powerful tool for the end users, it has been a great time in implementing the olution with lots of challenges we overcame in the past few weeks. The app is designed for simple features where there is always a chance to add more features or improve existing features. We welcome any suggestions to imporve our application.

VI. ACKNOWLEDGEMENT

We would like to thank our Mentor and Professor Rakesh Ranjan for guiding , mentoring and giving us this wonderful challenge to implement. We hope that we did out best to implement the solution and looking forward to him for further guidance and mentoring in future projects too.

VII. REFERENCES

[1] https://github.com/ryanmcgrath/twython

[2] https://github.com/mui-org/material-ui

[3]https://developer.twitter.com/en/docs/basics/authentication/overview/basic-auth

[4] https://flask-restful.readthedocs.io/en/latest/

[5]https://devcenter.heroku.com/categories/deployment

[6]https://aws.amazon.com/gettingstarted/tutorials/deploy-code-vm/