**Final Project Report**

**Digifesto**

**Team Cloud Squad**

**A person taking a selfie

Description automatically generated**

Rishab Macherla

Python-Backend & Cloud Developer

A person in a white dress

Description automatically generated with medium confidence

Gowthami Annem

Front-end & Cloud Developer

A person in a pink dress

Description automatically generated with medium confidence

Sagarika

Database Designer &

Cloud Developer

Please use this link to access the website: [Digifesto - #byteamcloudsquad](http://plsganeshfinalcsapp-env.eba-2iw3u89e.us-west-2.elasticbeanstalk.com/)

Use this link to access our docs - [link](https://drive.google.com/drive/folders/1GkpfY8yKExUSZ_-FUnN3wvsrzvdRgC2N?usp=share_link)

**Motivation:**

Elections, one of the most important valuable power every citizen has in any country. Citizens of any country will always be willing to elect a person as their leader whose intentions or motives help people to be free from any sort of problems they have. But how can a person who wants to participate in election know what exactly are the problems the citizens are facing? Is it logical and possible for anyone to visit every door and know their problems? No, right!!

Now, for this we are here with a solution called Digifesto – the digital manifesto which is the Leader’s strength. The main theme of this project is to eliminate the gap between common people and the people who wants to do something to the people of the nation. We here create a website which acts as a bridge between these 2 parties, so the people who directly want to convey something to their leader, can convey through this. Using this, every citizen can give their comments or compliments. The leader on the other end now has access to all the problems the people are facing, and he can also know the feedback of his people. Now, using all this available data of what problems people are facing, what people of thinking about their leadership these leaders can build a best manifesto. We all know Better the Manifesto more chance of being elected as a leader.

But is it possible for a person to read all of these i.e., the problems, comments, or compliments? Isn’t that a tedious task? To ease this task, we are making use of Machine Learning algorithms which help us analyze, clean the data. The idea here is, we in backend, first clean all the unwanted data. We, then use Natural Language Processing algorithm to remove all the unnecessary words from the statements the citizens have given. After this, we make use of Sentimental Analysis to analyze the given statement i.e., whether the given statement is positive statement or a negative statement. Based on this, we list them as positive and negative comments. Since it is not possible to read all these statements, we here develop Word clouds. These word clouds are built using the comments or complements the people gave. The words in this word clouds are highlighted depending on the frequency of the word being used. Let’s, take an example. If 100 people have complained regarding roads and 50 have complained regarding the sewage or anything, now since people have mostly complained about roads, the word road will be highlighted. So, using this word cloud the leader can conclude that bad roads are major issue people are facing and so he can list that out as a part of his manifesto. So, now if the leader guarantees that the road problems will mostly be resolved under his ruling, he has already won majority of the votes.

**System Architecture:**

This application involves 2 kinds of users, **Citizens**, other, **Leaders**.

Our application, Digifesto, acts as a bridge between these citizens and Leaders.

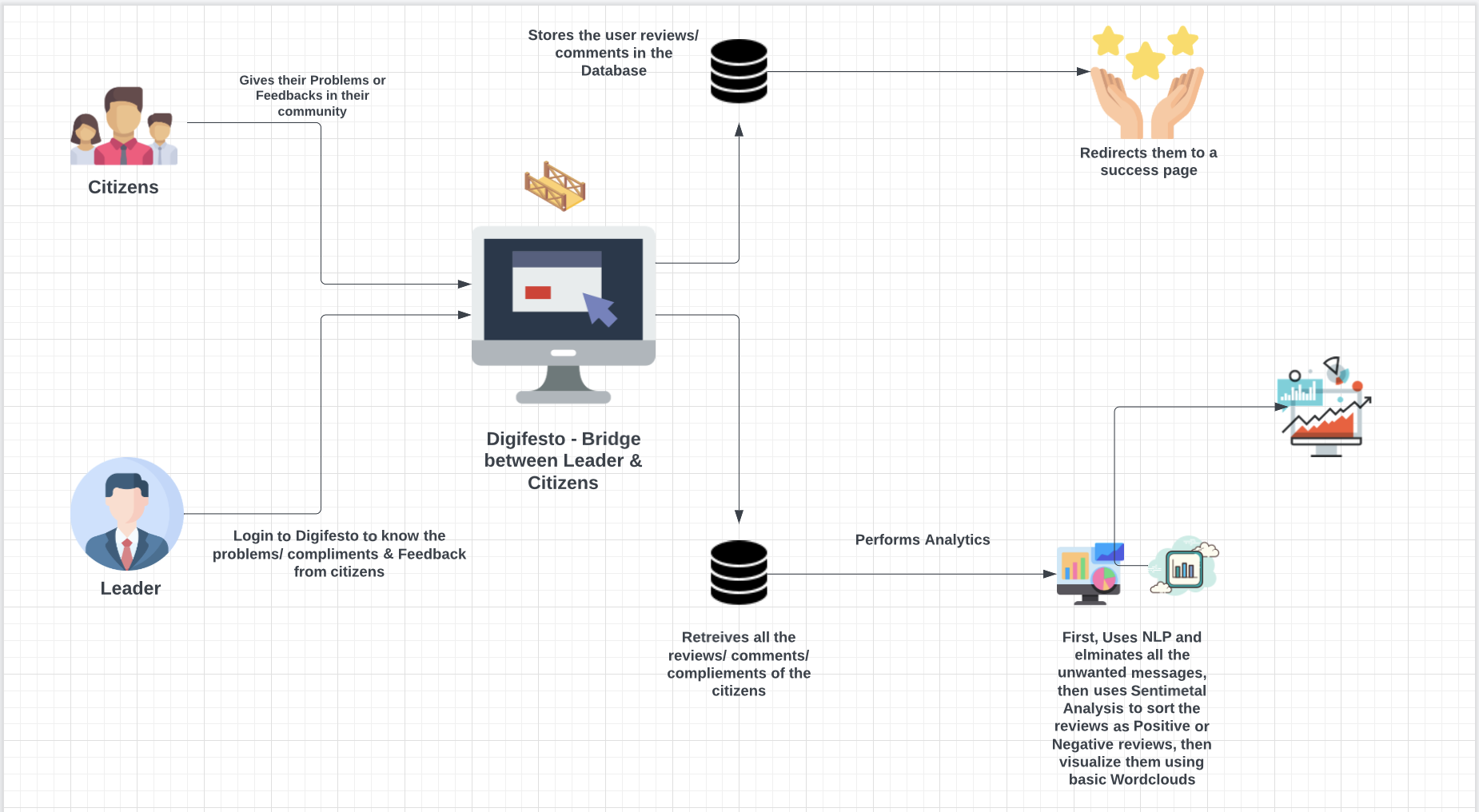
The users provide his problems, or compliments / comments about the leader/ Feedback and then submits it. Then, all these reviews are stored in the database. Once they are successfully stores, a user is redirected to a success page (for now).

When the Leader logs in, he uses his unique identifies (for now it is Password). At this instance of time, all the reviews which are stored in the database are pulled using Python in the back end and the entire analysis is performed i.e., these reviews first uses Natural Language Processing Algorithm and eliminates all the unwanted words from the reviews. Then these filtered reviews undergo Sentimental Analysis algorithm to sort the reviews as Negative & Positive Reviews. Then based on these 2 kinds of reviews, a word cloud separates for these 2 is created.

Now, these sorted reviews along with the word clouds, are displayed to the leader in his output page. The **advantage** of these word clouds is that, the size of the word in a word cloud depends on its frequency i.e., more the number of times a word is used, larger is the size of the word. So, these are used to make use of this advantage.

Let’s now consider and example. Let’s say, 10 users have used electricity, 6 users have used roads and 1 user have used sewage. Now, when the leader look at the word cloud, he finds that word roads have huge font, which means, without reading the reviews he can conclude that major problem to be solved is electricity and the next major problem is the roads. So, in his manifesto, he lists down in this order first electricity, then roads, then sewage. Now, in this case, if he guarantees and makes sure he has at least fulfilled the first thing in his manifesto, then he has already won majority of the votes. In this way, a better manifesto can be build, which not only helps the leader to win the election but also the problems of the citizens can be solved and the gap between these 2 is eliminated.

**Note**: This idea is not only restricted to Manifesto, but can also be implemented in many areas, making more and more features to it and making things better and solving all the problems with ease.

****

**Fig: System architecture of Digifesto**

**Features:**

Home page of Digifesto

**A blue rectangular object with white text

Description automatically generated with low confidence**

About Us – Our Team Cloud Squad

**A screenshot of a computer

Description automatically generated with medium confidence**

Our Team Members, their roles and responsibilities.

**A screenshot of a website

Description automatically generated with medium confidence**

Our Contact details in case of any queries

**A screenshot of a contact page

Description automatically generated with low confidence**

Citizens Page – Here the citizens provide their problems or reviews:

**(Note:** Being new to US and not being about the cities and areas, we made use of Indian communities & cities for demo)

**A screenshot of a survey

Description automatically generated with medium confidence**

Once the details are successfully stored in the Database, the citizen is given a success message**A white background with black text

Description automatically generated with low confidence**

**Login Page -** This is the Leader login page where the leader, provides his e-mail and password (for now later will be replaced with any of the unique identifier such as SSN, etc.). If he is new to this, then he has to create a new account by signing in.

A screenshot of a login screen

Description automatically generated

**Leader Page** – This is the leader page, where the leader, chooses his community to know the problems, feedback & compliments from the citizens.

**A screenshot of a computer

Description automatically generated with medium confidence**

**Leader Output page :** This is the leader output page, where all the reviews, etc. displayed after the elimination of unwanted messages, unwanted words and only the valid comments, problems, & compliments/feedbacks are listed after undergoing analysis.

**A screenshot of a computer

Description automatically generated with medium confidence**

From the above image, we can observe that, first, on the top all the reviews are revies are listed without being sorted. Then, since water problem (which is highlighted) is a negative review it is listed under **“Problems”,** and not only that it is also listed in the negative word cloud.

These are the word clouds which can help leader easily understand.

Note: For now, for this demo, we are using word clouds, later for the new update, we will be making use of some other better visualizations which can help leader in a much better way.

**A screenshot of a computer

Description automatically generated with low confidence**

For developing this application, we made use of different front-end, back-end technologies & for hosting we made use of AWS Beanstalk.

**Front-end Technologies used:**

* HTML
* CSS
* Bootstrap
* JavaScript
* jQuery

**Back-end Technologies used:**

* Python
* Django
* Natural Language Processing (NLP)
* Sentimental Analysis
* Word clouds

**Database**:

* SQLite

**Cloud Technologies used:**

* AWS Beanstalk
* S3 buckets

**Versioning Software:**

* GitHub

To refer the codes used for the development of this application, [click here](https://github.com/rm031011/CCFinalProject).