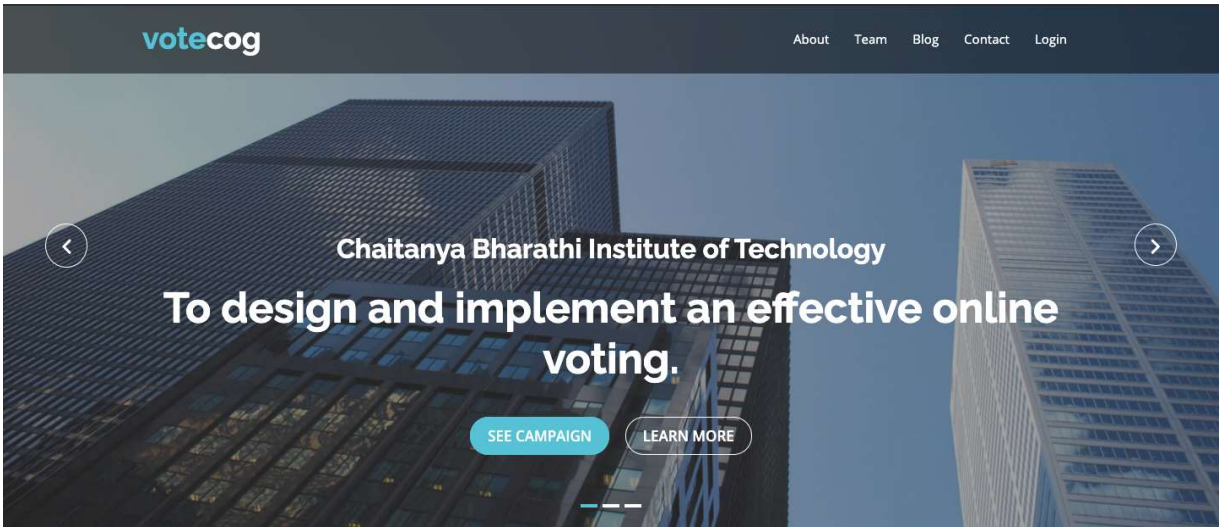


VOTECOG

ONLINE VOTING SYSTEM FOR STUDENT BODY ELECTIONS.

Report | Mini Project | 30-11-2020

VOTECOG

A Report Submitted in Partial Fulfilment for the Course Entitled Mini Project in V Semester for the Degree in Bachelor of Computer Science and Engineering.

By,

Komali Beeram – 1601 18 733 067

BE ¾ Computer Science and Engineering

G Satvika Reddy – 1601 18 733 079

BE ¾ Computer Science and Engineering

Under the Guidance of,

Mr. B Sateesh

Assistant Professor

Chaitanya Bharathi Institute of Technology

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY (A)

Affliated to Osmania University

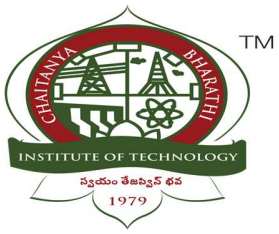
Department of Computer Science and Engineering

Chaitanya Bharathi Institute of Technology

Gandipet, Hyderabad -500075

Website: www.cbit.ac.in

November 2020



CHAITANYA BHARATHI
INSTITUTE OF TECHNOLOGY (A)
Affliated to Osmania University

CERTIFICATE

This is to certify that the project titled "Votecog", submitted to Department of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, in partial fulfilment of the requirements for the B.E V Semester course titled Mini Project, is a bonafide record of work carried out by Komali Beeram (1601 18 733 067) and G Satvika Reddy (1601 18 733 079), during the academic year 2020-2021 under our guidance and supervision.

No part of this project has formed the basis for the award previously of any other degree, diploma, fellowship, or any other similar title.

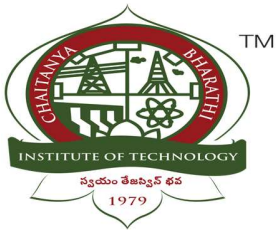
Mentors

Sri. B Sateesh
Assistant Professor
Department of Computer Science and Engineering,
Chaitanya Bharathi Institute of Technology.

Smt. I Srujana
Assistant Professor
Department of Computer Science and Engineering,
Chaitanya Bharathi Institute of Technology.

Supervisor

Sri. B Sateesh
Assistant Professor
Department of Computer Science and Engineering,
Chaitanya Bharathi Institute of Technology.



CHAITANYA BHARATHI
INSTITUTE OF TECHNOLOGY (A)
Affliated to Osmania University

DECLARATION

We, here by, declare that the project report entitled "Votecog" submitted to Department of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, in partial fulfilment of the requirements for the B.E V Semester course titled Mini Project, is a bonafide record of work carried out by Komali Beeram (1601 18 733 067) and G Satvika Reddy (1601 18 733 079), during the academic year 2020-2021 under the guidance and supervision of Mr. B Sateesh and Smt. I Srujana. We further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Hyderabad

30 November 2020

Komali Beeram
1601 18 733 067
BE ¾ Computer Science and Engineering
Chaitanya Bharathi Institute of Technology

G Satvika Reddy
1601 18 733 079
BE ¾ Computer Science and Engineering,
Chaitanya Bharathi Institute of Technology.

ACKNOWLEDGMENT

The success and the outcome of this project required a lot of guidance and assistance from many people and sources and we are extremely fortunate to have received this all along the completion of the project.

We would like to express our special thanks of gratitude to our supervisor, Mr. B Sateesh, Assistant Professor, Department of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, for his invaluable guidance, moral support and patience while suggesting the outlines of this project and in meeting the deadline. We thank for his overall support.

We would also like to express our sincere indebtedness towards our mentors Mr. B Sateesh and Smt. I Srujana for their approval of the project, and time to time counselling and advices.

We also convey our sincere thanks to the Head of Department of Computer Science and Engineering, Dr. Y Rama Devi, for her for constant feedback and assistance.

We also thank our college, Chaitanya Bharathi Institute of Technology, for giving us the opportunity to work on an innovative idea under the course Mini Project.

At last, we would like to thank our families and friends for their constant support and guidance from time to time in making this project well above simplicity and into something concrete, despite their busy schedules.

ABSTRACT

In the new era of advanced technology where online system boosts work speed, reduces mistakes, and promotes the generation of accurate results, having manual voting system becomes a misfortune. A public voting system constitutes the backbone of a democracy where the students must choose/elect their council representative. CBIT's student council currently uses a manual voting system, which causes several kinds of problems. Due to this ballot-based voting system, some problems are faced by voters before or during elections and others are faced by the administration before and after the voting. An online system, which involves procedures like vote casting, vote counting, declaring results, campaigning etc. would constitute a good solution to replace current system. Moreover, the online voting system will also reduce any malpractices taking part during elections.

TABLE OF CONTENTS

List of figures

INTRODUCTION

Background

University student leaders are the core link between the students and the university administration. These leaders are therefore elected democratically to represent the interests of the students as per the university act. It is always an expectation of every student that elections be held fairly, and results computed accurately. In the previous elections, there have been challenges in the turnout of voters due to some challenges they face on the voting day. Initially the students were expected to queue and cast their votes in the ballots. The current system does not verify and account for the persons to vote since no voting registration is done prior. This has been bringing some loopholes in that even a student who is not in university can queue and vote as long as he/she has a student identification Card. The current system does not also tell the number of expected voters since they rely on the population of the student of which not all students are interested in these elections. This is the main challenge to the voters and officials of election commission. Candidates are expected to reach voters through campaigns, rallies and debates which has been a challenge for a while due to the busy environment in the university. Many students do not attend rallies because of their busy schedules and, it's hard for candidates to find students in their houses since availability is an issue. This mode of campaigning has been yielding less fruits and thus voters would cast votes with either no or less knowledge about the candidates and their manifestos. When it gets to counting, a lot of time man powers are consumed following the large population of voters. With the nature of human being not being so diligent, a lot of flaws are found which brings a lot of chaos from the stakeholders. Following these challenges, we saw it good to come up with a system that could curb these problems and speed up the election system to ensure free and fair elections. When a system that is based on pens and ballot papers is used at a large population, the results can be ambiguous and that questions the intelligibility of the system used. Hand counting votes is time consuming especially at a turnout of many voters and many positions being voted for. In a case of disabled or duty-bound people, they struggle to cast their votes. The system makes it easy for them since they can vote at their comfort of their places. This system also curbs the chances of the manipulation of results from influencing authorities and thus generate transparency to the highest levels.

Current system

The current system is a manual system where everything is done by using manpower. The aspiring candidates apply through their departments for the various posts of interests. They are therefore vetted by the senior authorities and nominated for the posts. Afterwards they are given a duration of campaign where they must sell their policies and agendas to the voters and win their votes. These campaigns are done through class to class campaigns where they walk in between lectures asking for votes and also some one to two rallies that are expected to tell their manifestos and defend themselves for the posts. The rallies are always a challenge since the supporters of the opponents may interfere with the speech of a candidate or just cause chaos or crannies to prevent others from listening to the candidate. The debates organized in conjunction to the election are always ineffective since they provoke conflicts and chaos from the opposite possible teams.

Voting is done manually using pen and ballot papers where a student is supposed to tick or put a mark alongside the candidate of choice. This many times leads to so many spoilt votes due to ignorance or violation of rules of the students who are hungry to vote. The students make long queues to cast their votes as voting is done one student at a time. Counting of votes is done manually with the candidate's agents witnessing the process. The candidate with the majority of votes is declared winner and sometimes once the candidate with major votes is noticed, the counting process is stopped. How then will the other candidates know by how many votes have they lost? This is the reason for our proposed system. The system is so slow, tiresome and with a lot of loopholes as there may be some essence of manipulation of results from the higher authorities in favor of their preferred candidates who may not be the choice of the people.

Problem statement

The current system of election does not take record of voters hence gives chance to any person to vote as long as they have the university identification cards. Some students may not be eligible for the process since they may not be in the university system on different reasons. Interaction between voters and candidates has been minimal since they only interact in rallies which are done once and may not be enough for all students to know who the candidates are and what the candidates have for them. Senior authorities may exploit and manipulate the votes in favor of their preferred candidates which tempers with the expected free and fair elections. The current system consumes a lot of time since users must queue in order to vote and also counting is hand counting which takes a lot of time and manpower. The proposed system will provide online candidate registration forms for students where students will register as candidates and be allowed to log in as either students or candidates. Each registered user will have a password to log in.

The proposed system will provide an interactive platform where voters and candidates will interact and thus candidates perform and analyze their campaigns. The system will also perform some sort of tallying where results and statistics on the expected election will be shown and updated properly. The system will allow preliminary voting and the results will be graphically represented in percentage. This system will also allow the candidates to be liked by users and the most liked candidate is the most popular. The system will compute and give the election results for all the posts and provide reports for the whole election process.

Proposed System

The proposed system will provide login credentials to only those students who are in the university with which they can interact with the system. The student details will be saved in the accounts database. The users will be able to log in as either normal student or delegate or candidate. Delegates details are saved in the admin database while candidates details are saved in the persons database.

The proposed system will also provide interaction platforms for both the voters and the candidates where they will interact and discuss matters elections. The candidates shall therefore perform their campaigns and answer the possible questions from the voters on the platform. The candidates will be allowed to pose their various agendas and manifestos and defend them on interrogation of the delegates. Candidates will interact with delegates and respond to their queries accordingly.

The system will be able to perform some sort of tallying before and after voting. The results and statistics on the election will be shown and be updated properly and instantly. The system shall also allow preliminary voting where the delegates are able to vote and results are displayed openly to everyone. The students and delegates will be able to like the various candidates and the most liked candidate will be the most popular. The system will display

the results of likes and votes and even percentage of victory. The proposed system will therefore compute the election results for all the posts voted and compute the most popular candidate and the least popular candidate. The system will also compute reports for the whole election process.

Objectives

The following are the objectives of the proposed system:

- To design, develop and implement an efficient, user friendly, interactive we based online voting system for student body elections.
- To develop a system that will capture candidates and voters' details.
- To develop a system that will facilitate online voting.
- To develop a system that will facilitate voters and candidates' interactions.
- To develop a system that will generate analysis of the election process.
- To develop a system that is scalable, accurate, efficient, secured, and fast.

LITERATURE REVIEW

This section documents the available relevant literature concerning the problem domain. The implication was that we devoted sufficient time to reviewing research already undertaken on related problems. This was done to find out what data and other materials are already available from earlier research and identify gaps that the present research may fill.

Elections

Election is the process that gives the citizens the rights to select candidates to represent them in a democratic pattern. Election deals with the democracy and freewill of citizens, for this reason voting process is very critical and sensitive process, therefore election implementation must serve many requirements to deliver a trustworthy election. These requirements can be defined as user conventions requirements and delivery of secure voting process requirements.

Due to the fast development of network technology the world is going toward the use and implementation of the e-technology in every aspect of our life including e-governments. Online voting becomes one of these technologies. Online voting refers to the use of hardware and software to establish an electronic system, useful in voting process, by generating an electronic ballot that replaces the paper ballot. E-voting was introduced by e-governments especially in Europe in order to serve voting convention by providing remote system so the voter can cast his/her vote whenever and wherever he/she can. These systems will increase voter's participation and will speed up the votes counting. Introducing remote voting technique over the internet (e-voting) will serve voter's convention. The main idea of this technology is to speed up the ballot counting and increase voters' participation by providing remote voting process and social interaction platforms.

Types of voting

Voting is a process at the heart of a democratic society. There is a wide variety of different voting systems that are based on traditional paper ballots, mechanical devices, or electronic ballots.

Ballot papers

Handwritten paper ballots were first used in Rome in 139 BCE, and their first use in America was in 1629, to select a pastor for the Salem church. These early paper ballots offered only modest voter privacy and they were fairly easy targets for various forms of election fraud. The modern system of election using paper ballots was first used in 1858 in Australia. The great Australian innovation was to print standardized ballots at government expense, distribute them to the voters at the polling places, and require that the voters vote and return the ballots immediately. Today, the security against election fraud this provides seems obvious, but in the 19th century, it was not obvious to most observers, and it was not until 1888 that this ballot was used in the United States. A

Figure 1 Ballot Papers

properly administered Australian paper ballot sets a very high standard, assuring voter privacy, preventing voters from revealing how they voted, and assuring an accurate and impartial count.

Lever voting machines

Lever voting machines were first used in 1892 in New York and were slowly adopted across the country. They eliminate most of the approaches to manipulating the vote count that were endemic a century ago, and they can easily be configured to handle a complex general election ballot. Lever voting machines offer excellent voter privacy, and the feel of a lever voting machine is immensely reassuring to voters! Unfortunately, they are immense machines, expensive to move and store, difficult to test, complex to maintain, and far from secure against vote fraud. Furthermore, a lever voting machine maintains no audit trail. With paper ballots, it is possible to recount the votes if there is an allegation of fraud. With lever voting machines, there is nothing to recount! In effect, lever voting machines were the "quick technological fix" for the problems of a century ago; they eliminated the problems people understood while they introduced new problems. Because they are expensive to test, complete tests are extremely rare. The mechanism is secure against tampering by the public, but a technician can easily fix a machine so that one voting position will never register more than some set number of votes, and this may not be detected for years.

Figure 2 Lever Voting Machine

Punched cards

The first new technology to effectively challenge lever voting machines was the now infamous Votomatic voting machine. Punched card data processing dates to the 1890's, but IBM did not introduce the Votomatic punched card voting system until 1964. The Votomatic ballot and the more recent mark-sense ballot both represent a return to the Australian secret ballot, but with the added benefit of automated and impartial vote count produced using tabulating machinery. With this return to paper ballots, we gained the ability to recount the vote in the event there is a challenge, but we also introduce the question of how to interpret marginal votes. From a legal perspective, a ballot is an instrument, just like a deed or a check. When the ballot is deposited in the ballot box, it becomes anonymous, but just prior to the moment when the ballot is deposited, it ought to be possible to hand the ballot to the voter and ask "does this ballot properly represent your intent?". Votomatic punched card ballots fail this simple test! While the ballot is in the Votomatic machine, the voter can punch holes in it but is unable to see the ballot itself. Once removed from the machine, the voter can see the holes, but without the ballot labels printed on the machine, the voter is unable to tell what those holes mean.

Figure 3 Punched cards

Optical mark sense ballots

Optical mark-sense voting systems were developed in the early 1970's by American Information Systems of Omaha, alternately in competition with and in cooperation with Westinghouse Learning Systems of Iowa City. The latter was the licensee of the University of Iowa's patents on the optical mark-sense scanning machine. Essentially the

Figure 4 Optical mark sense ballots

only advantage of mark-sense technology over punched card technology is that it uses marks on a printed paper ballot. This is an important advantage! This means that no special machines are required to vote on the ballot, it means that, with proper ballot design, a voter can easily verify that the markings on the ballot exactly convey his or her intent, and it means that, during a hand recount, no special expertise is required to interpret the intent of the voters. Unfortunately, the first generation of optical mark-sense voting machines was extremely sensitive to the pen or pencil used to mark the ballot, and to the exact details of the mark itself. As a result, early machines, including many still in use today, had real difficulty distinguishing faint deliberate marks from smudged erasures, and they tended to have mark sensing thresholds that required a dark mark. The newest generation of optical mark-sense readers uses visible wavelength image processing technology instead of simple infrared sensors to read the marks. Many of the more recent offerings use either FAX machine scanning mechanisms or computer page-scanning devices to obtain the image of the ballot, and they operate by finding each marking target before they search the target for acceptable marks. Such machines can easily ignore relatively dark smudged erasures while catching relatively faint deliberate marks.

Electronic voting

This is a voting system where the recording, casting, and counting of votes involves information and communication technology. The main principle of e-voting system is the replica of the regular voting system as much as possible it is compliant with the election legislation and principles and be at least secure as the regular voting. In a nutshell, e-voting strives to be uniform and secret, only eligible persons are to be allowed to e-vote and a voter should only cast one vote and the collections are to be secure, reliable and accountable. This system gives loopholes to election theft and manipulation of votes especially during the collection of votes. The proposed research works on filling this gap. The process of this system is ambiguous in a manner that a voter has to register and keep confirming whether his/her details are in the system and on the voting day, voters have to cast their votes in the ballot. It is proved to be challenging for the system to accommodate the disabled and multilingual voters hence it is also time consuming. for the e-voting system to function properly, it should ensure error-free and robust electronic voting over the internet which has been a difficult for this system hence it could not be implemented in most of the institutions worldwide.

Figure 5 Electronic voting machine

Online voting

This is just like electronic voting but only that it is web-based system. In this paper a new easy to use, secure and transparent online voting system is proposed. The new scheme can be easily used by colleges and universities worldwide. The new scheme most notable allows voters to interact, participate in campaigns and political rallies virtually by use of a web-based system.

Identified gaps

- Lack of transparency in the election systems.
- Lack of robust and error free systems for election process.

DEVELOPMENT METHODOLOGY

The smooth functioning of the project depended on the development of multiple components; the following sections will briefly discuss each one. The development methodology used is the waterfall. This included the analysis, design, implementation, and the testing steps.

Analysis

In this case we analyzed the requirements, and fully understood the problems. Analysis was conducted on the current systems failures and strengths. This allowed a better understanding of the expected improvements. Further analysis was also conducted on the problem definitions to clearly understand what to tackle. This phase is usually accompanied by documentation for each requirement, which enabled us to review it for validation.

Logical design

Logical design characteristically looked at the intended system from a logical perspective without considering physical requirement. The project needed a logical design that modelled the flow of data and information through the system from input to output. Logical design also modelled the security checks that the system will be using as well as the formats for all data items in the system.

Physical design

The physical design is concerned with how the physical architecture of the entire system interacted to achieve its objectives. It modelled the user interfaces, the server architecture, and the database models.

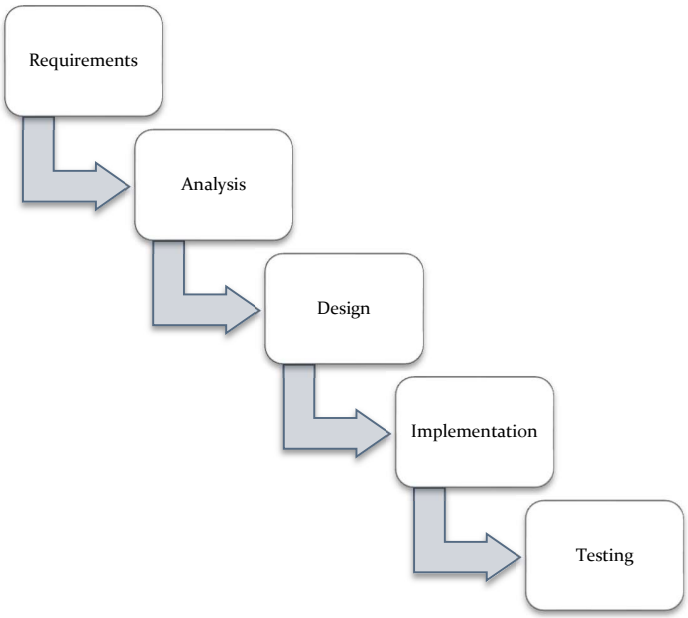


Figure 6 Waterfall Development Methodology

IMPLEMENTATION

Once the designs are deemed to be viable, technical implementation begins. Implementing the project was the toughest part as all the coding was done in this phase. Being that the project serves only the web platforms, coding took place in three phases.

Database coding phase

The database we use is the inbuilt django SQLite, for the concised database we used it was sufficient. Furthermore, on large scale we can use Mysql. During the course of our project we used tables that were absolutely necessary and we made use of foreign keys to help connect the databases and use them in our code more efficiently.

Firstly, the *Account* model was used to allow access all the registered members under our platform i.e admin, candidates and user. It contains the following fields which are absolutely necessary for our prototype. Once a voter votes his ability to vote will automatically be taken away.

```
    email = models.EmailField(verbose_name="email", max_length=60, unique=True)
    username = models.CharField(max_length=30, unique=True)
    date_joined = models.DateTimeField(verbose_name='date joined', auto_now_add=True)
    last_login = models.DateTimeField(verbose_name='last login', auto_now=True)
    is_voter = models.BooleanField(default=True)
    is_admin = models.BooleanField(default=False)
    is_staff = models.BooleanField(default=False)
    is_active = models.BooleanField(default=True)
    is_superuser = models.BooleanField(default=False)
    voted = models.CharField(max_length=30, default="no")
```

Figure 7 Accounts model

The person model helps us keep a track of all the candidate requests that came in along with their status of their ability to vote. Once a voter votes for a candidate, the vote field will increment by 1, thus keeping a track on the number of votes that were casted.

```
   user = models.ForeignKey(Account,on_delete=models.CASCADE)
    name = models.CharField(max_length=130)
    branch = models.CharField(max_length=8, choices=DEPT_CHOICES, default='cse')
    year = models.CharField(max_length=6, choices=YEAR_CHOICES, default='1st')
    bio = models.TextField(blank=True)
    status= models.CharField(max_length=130,default="pending")
    published_date = models.DateTimeField(default=timezone.now)
    vote=models.IntegerField(default=0)
```

Figure 8 Candidate model

The Post model keeps a track on all the posts that were used during the period of the election campaign. We also make sure we take the Account table as our foreign key so that we can easily map the owner of the post

```
class Post(models.Model):
    # user = models.ForeignKey(Account, on_delete=models.CASCADE)
    author = models.ForeignKey(Account,on_delete=models.CASCADE)
    title = models.CharField(max_length=200)
    text = models.TextField()
    created_date = models.DateTimeField(default=timezone.now)
    published_date = models.DateTimeField(blank=True, null=True)
    liked = models.ManyToManyField(Account, default=None, blank=True, related_name='liked')
```

Figure 9 Campaign post model

As the name suggests the Like and Comment models are used to keep a track on all the likes and comments on the post including the identity of the person to be safe of any misuse of this feature by the candidates and voters. These models keep a track on the post on which the comment or like was made as well.

```
class Like(models.Model):
   user = models.ForeignKey(Account, on_delete=models.CASCADE)
   post = models.ForeignKey(Post, on_delete=models.CASCADE)
   value = models.CharField(choices=LIKE_CHOICES, default='Like', max_length=10)

   def __str__(self):
       return str(self.post)

class Comment(models.Model):
   post = models.ForeignKey(Post, on_delete=models.CASCADE,
related_name='comments',null=True, blank=True)
   user = models.ForeignKey(Account,on_delete=models.CASCADE,null=True, blank=True)
   content = models.TextField(max_length=250)
   date_added = models.DateTimeField(auto_now_add=True)

   def __str__(self):
       return str(self.content)
```

Figure 10 Voters response to campaigns model

Machine Learning coding phase

Sentiment analysis is a common NLP task, which involves classifying texts or parts of texts into a pre-defined sentiment. You will use the Natural Language Toolkit (NLTK), a commonly used NLP library in Python, to analyse textual data. We used the negative and positive tweets to train your model on sentiment analysis later in the tutorial. The tweets with no sentiments will be used to test your model.

Noise is any part of the text that does not add meaning or information to data. Noise is specific to each project, so what constitutes noise in one project may not be in a different project. For instance, the most common words in a language are called *stop words*. Some examples of stop words are “is”, “the”, and “a”. They are generally irrelevant when processing language, unless a specific use case warrants their inclusion.

Normalization in NLP is the process of converting a word to its canonical form. Normalization helps group together words with the same meaning but different forms. Without normalization, “ran”, “runs”, and “running” would be treated as different words, even though you may want them to be treated as the same word.

We used the Naive Bayes classifier in NLTK to perform the modelling exercise. The model requires not just a list of words in a tweet, but a Python dictionary with words as keys and True as values.

```
from nltk.stem.wordnet import WordNetLemmatizer
from nltk.corpus import twitter_samples, stopwords
from nltk import NaiveBayesClassifier
```

Figure 11 Modules required for sentiment analysis on campaigns

This code attaches a Positive or Negative label to each tweet. It then creates a dataset by joining the positive and negative tweets.

By default, the data contains all positive tweets followed by all negative tweets in sequence. When training the model, you should provide a sample of your data that does not contain any bias. To avoid bias, you’ve added code to randomly arrange the data using the shuffle() method of random.

Finally, the code splits the shuffled data into a ratio of 70:30 for training and testing, respectively. Since the number of tweets is 10000, you can use the first 7000 tweets from the shuffled dataset for training the model and the final 3000 for testing the model.

In this step, you converted the cleaned tokens to a dictionary form, randomly shuffled the dataset, and split it into training and testing data.

```
positive_tokens_for_model = get_tweets_for_model(positive_cleaned_tokens_list)
negative_tokens_for_model = get_tweets_for_model(negative_cleaned_tokens_list)

positive_dataset = [(tweet_dict, "Positive")
                    for tweet_dict in positive_tokens_for_model]

negative_dataset = [(tweet_dict, "Negative")
                    for tweet_dict in negative_tokens_for_model]

dataset = positive_dataset + negative_dataset

random.shuffle(dataset)

train_data = dataset[:7000]
test_data = dataset[7000:]
```

Figure 12 Training and testing on dataset

Web coding phase

The project is coded in HTML, CSS, JavaScript and bootstrap for the web platform.

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web-server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearances of the document.

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

Bootstrap is a free and open-source front-end library for designing websites and web applications. It contains HTML and CSS based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with the front-end development only.

DESIGN

Upon completion of full implementation, testing occurred before it got into public consumption. We used the design documents, personas, and user case scenarios to run comprehensive tests including the Components testing and on the finished applications.

System Design

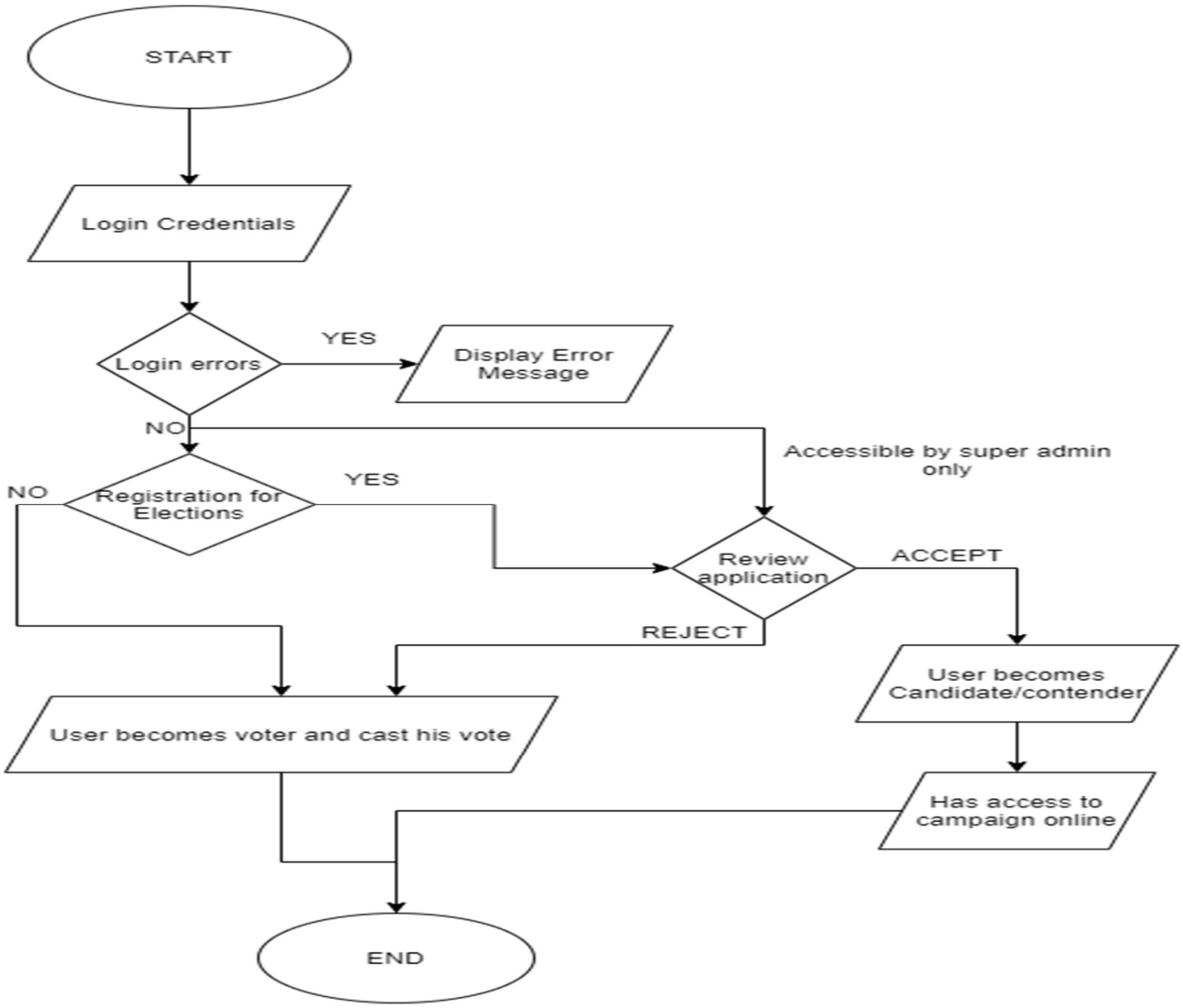


Figure 13 UML diagram

System requirement

The following are the requirements needed for the system. The system runs only on the web platform and therefore the requirements were met to enable the expected performance.

- 32- or 64-bit operating system.
- Web browser: Above chrome 12, Firefox 14, Internet explorer 7, Safari 2, Edge 17
- JavaScript supported and enabled for front end logic execution.
- Single/Multi Core + 1GHz and above Microprocessor.
- RAM: 512MB and above.
- Internet Access: 1MBps downlink, 512KBps Uplink for WAN Online Functionality.
- Python version above 3
- Django version 3.0.0

User requirement

The user should be literate in English and eligible to use the system. The user should also access internet to interact and use the system since it’s a web-based application.

TESTING

Unit Testing

Unit involves testing software with a small piece of source code (unit, component, and/or function) of the same software. During performing tests, some hypotheses were made, and the testing was then determined if true or false. This way, the developer was able to check whether a unit behaves as intended or whether a unit corresponds to the design specifications. All the sources used in unit testing were created by the developer as a part of software development. The following unit tests were performed to ascertain functionality.

Navigation Test

This test verifies if the user can navigate the site and access all URLs. Testing a login scenario.

Authentication Tests

This test verifies the username and password to access Votecog web application.

Interface Testing

Interface Testing was performed to evaluate whether systems or components pass data and control correctly to one another. It was also used to verify if all the interactions between these modules are working properly and errors are handled properly. To perform the interface tests, the developer created a checklist that outlined all the functional requirements of the system and the various test cases to assess them.

Authentication Tests

To verify and authenticate user using username and password. On successful authentication, user is directed to homepage. The valid input for authentication is well formatted credentials.

Updating Tests

To update profiles for candidates once the delegate accepts their application to contest in elections. The output to this test is edited details of the user to candidate with extra privileges will be reflected/displayed.

Voting Tests

Voting of Candidates. The output to this test is pop up alert displaying success in voting.

Usability Testing

The table below summarized tests that were performed to ascertain the usability and experience of users while interacting with the system.

ELEMENT	OUTPUT
Flow from start to finish	Yes
Feedback from actions performed	Instant Feedback
Seamless Navigation	Yes
Performance	Optimal
Failure or crashes	None
Runtime error messages	None
Slow or delayed loading	Acceptable

Integration Testing

This checks whether the various components of the system are integrated and working in sync. All the screens, functions, stores, data tables and other modules were connected with seamless interfacing. All the required outputs were produced successfully as expected from the systems and all inputs were validated and stored in the correct formats.

RESULT

The web pages of the application as per the design of UI are attached below.

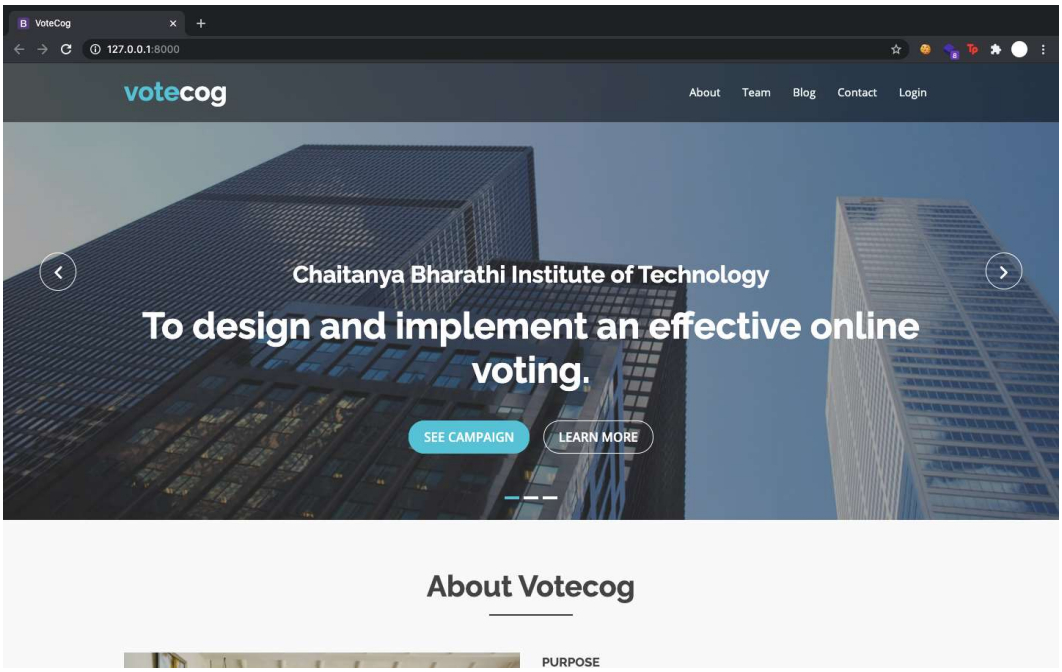


Figure 14 Home-page

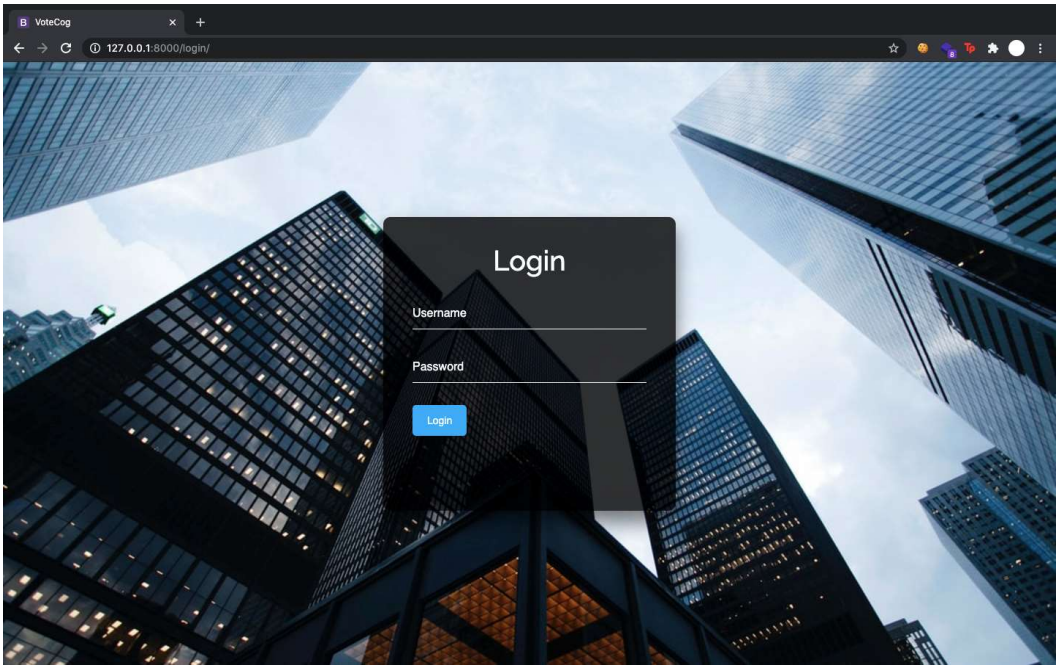


Figure 15 Login Page

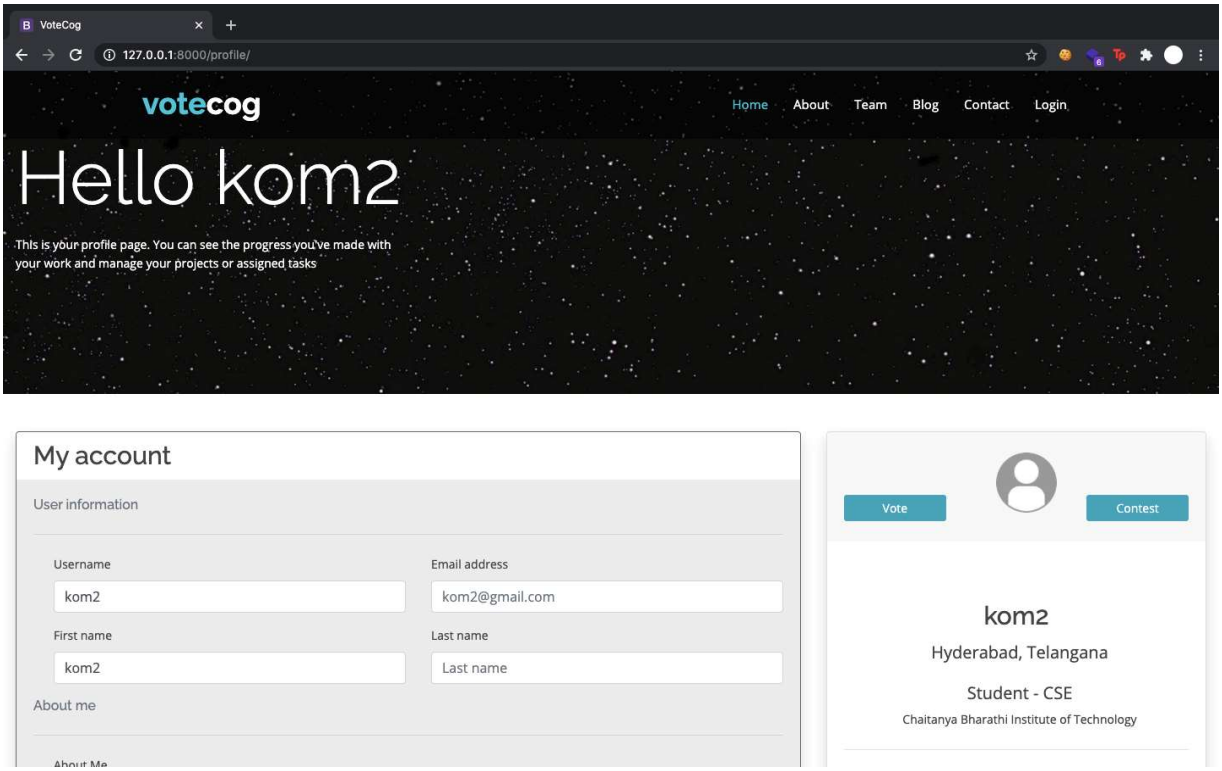


Figure 16 Voter profile page

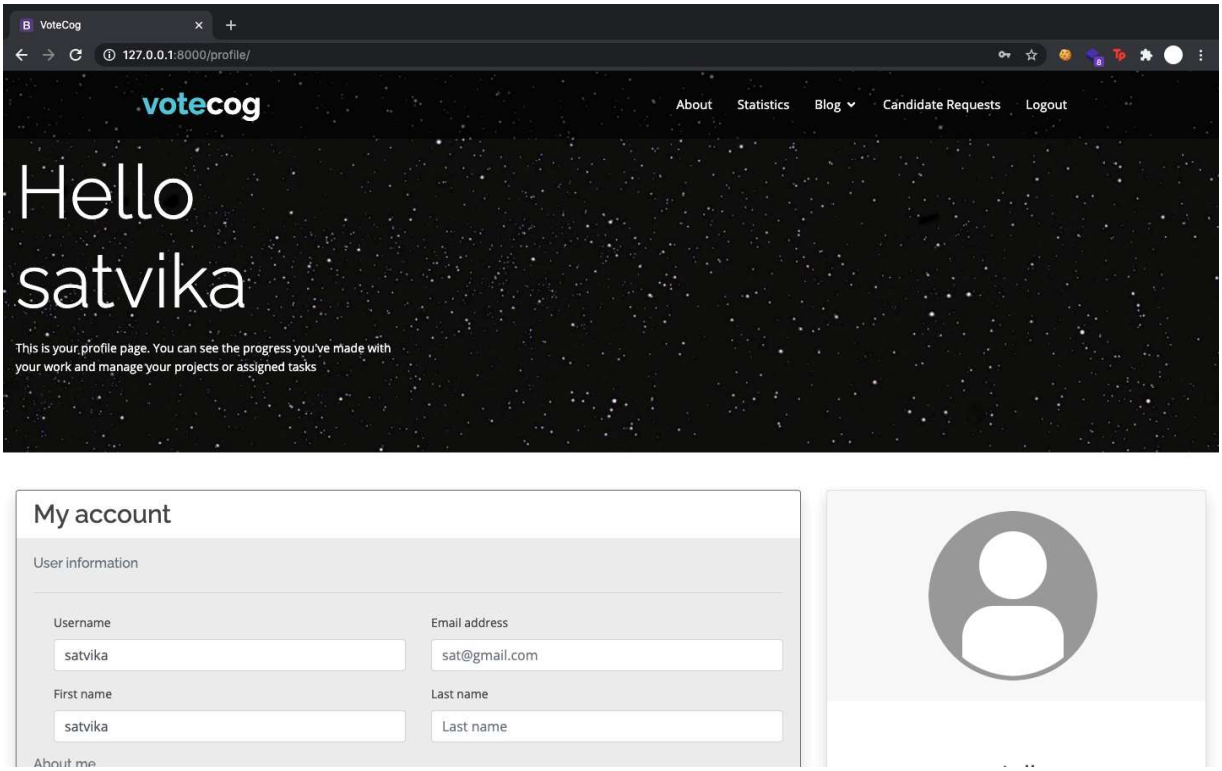


Figure 17 Candidate profile page

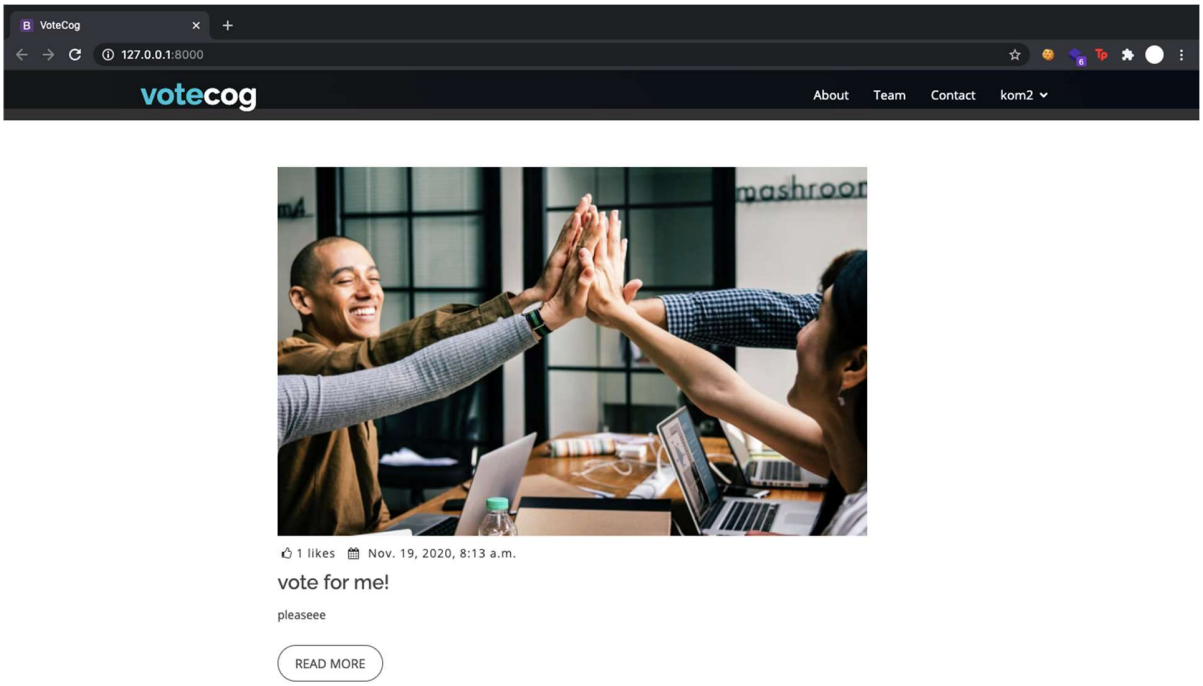


Figure 18 Online Campaign posts

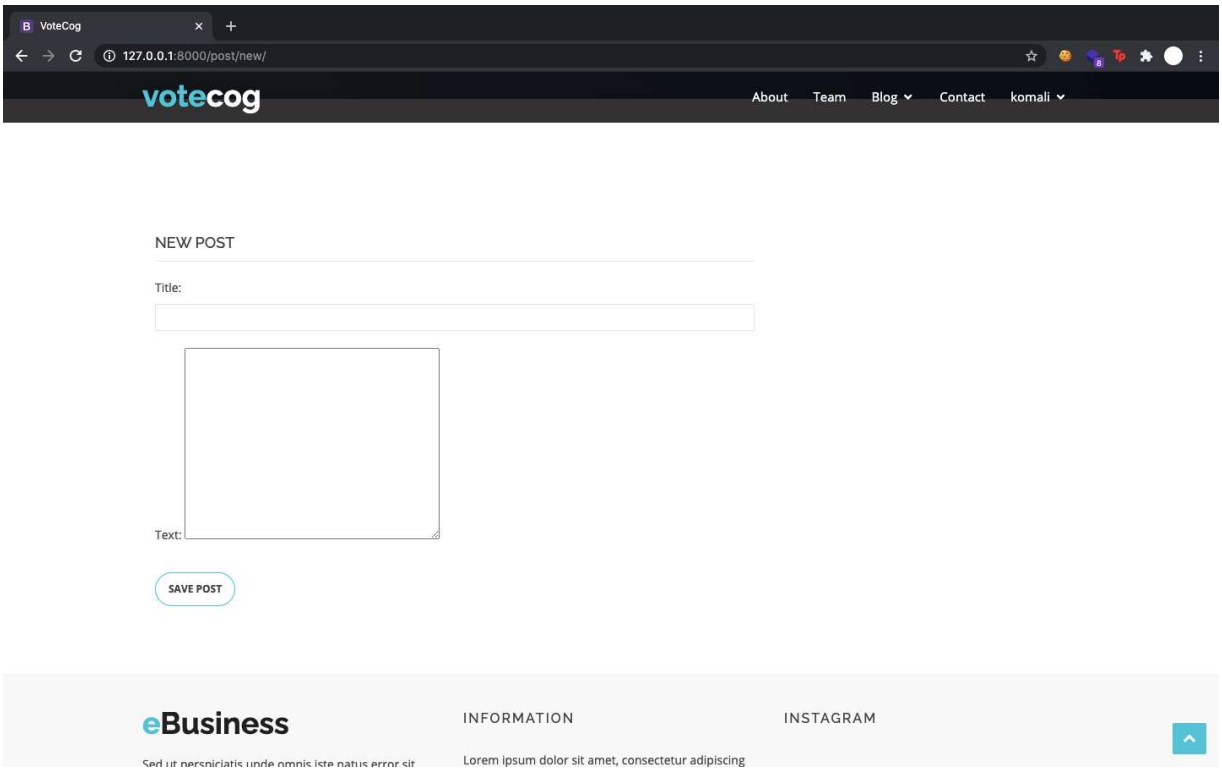


Figure 19 Add new campaign post

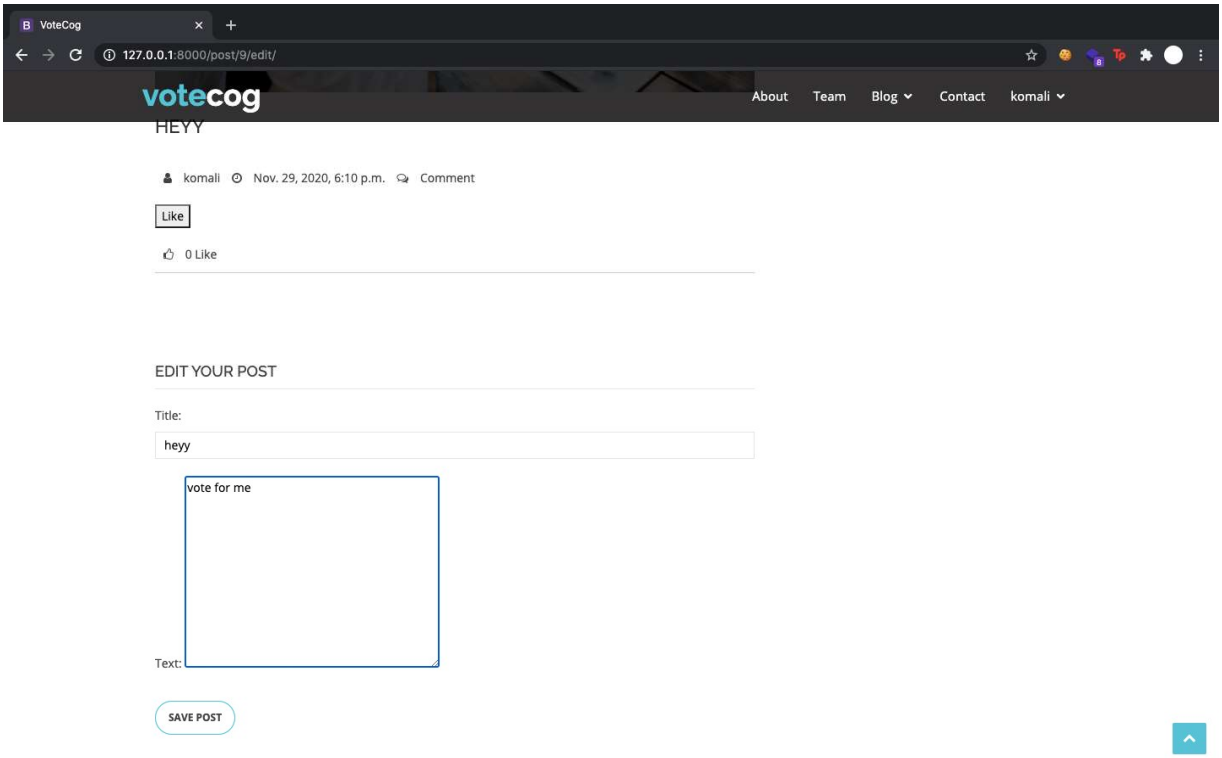


Figure 20 Edit campaign post

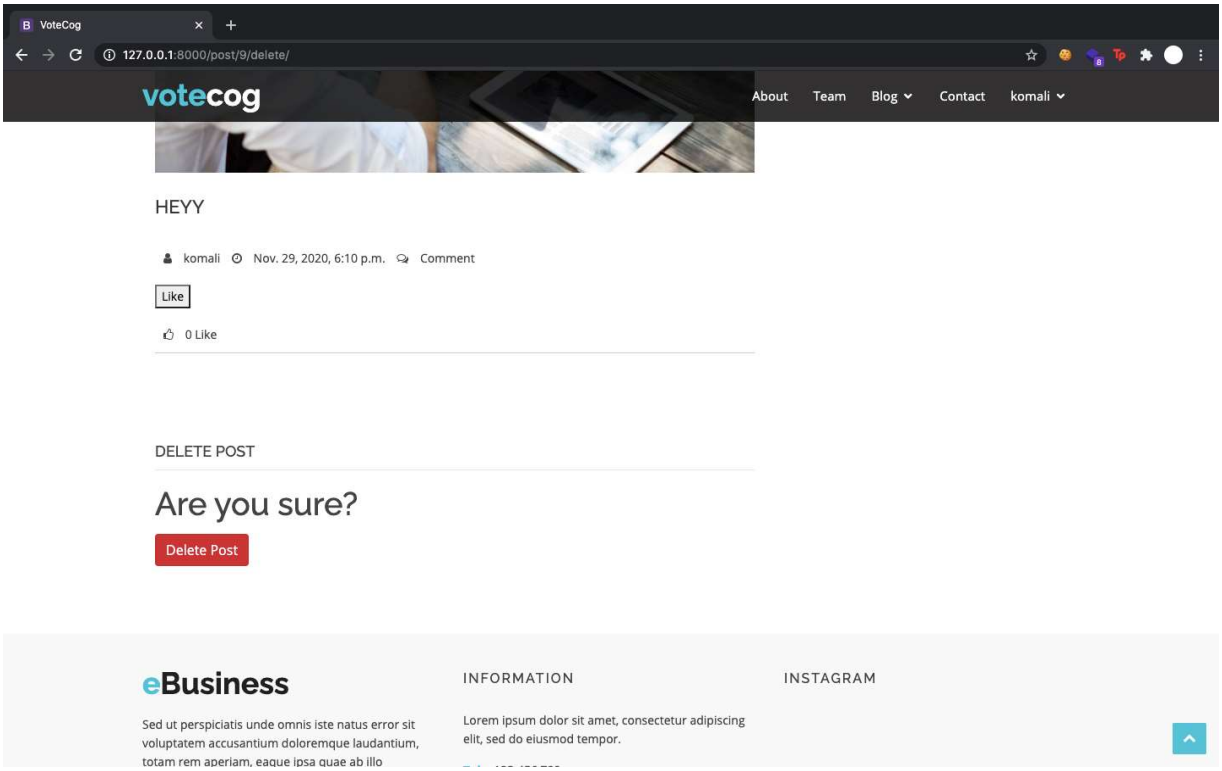


Figure 21 Delete campaign post

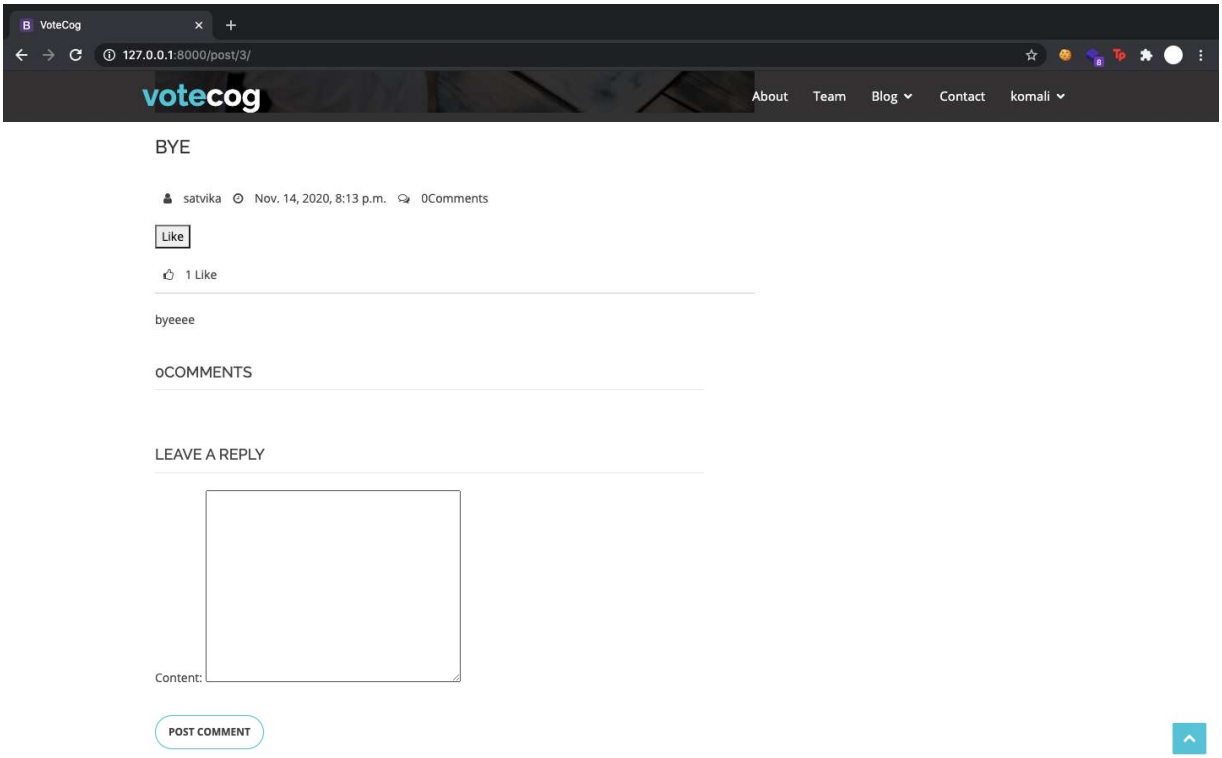


Figure 22 Comment and like campaign post

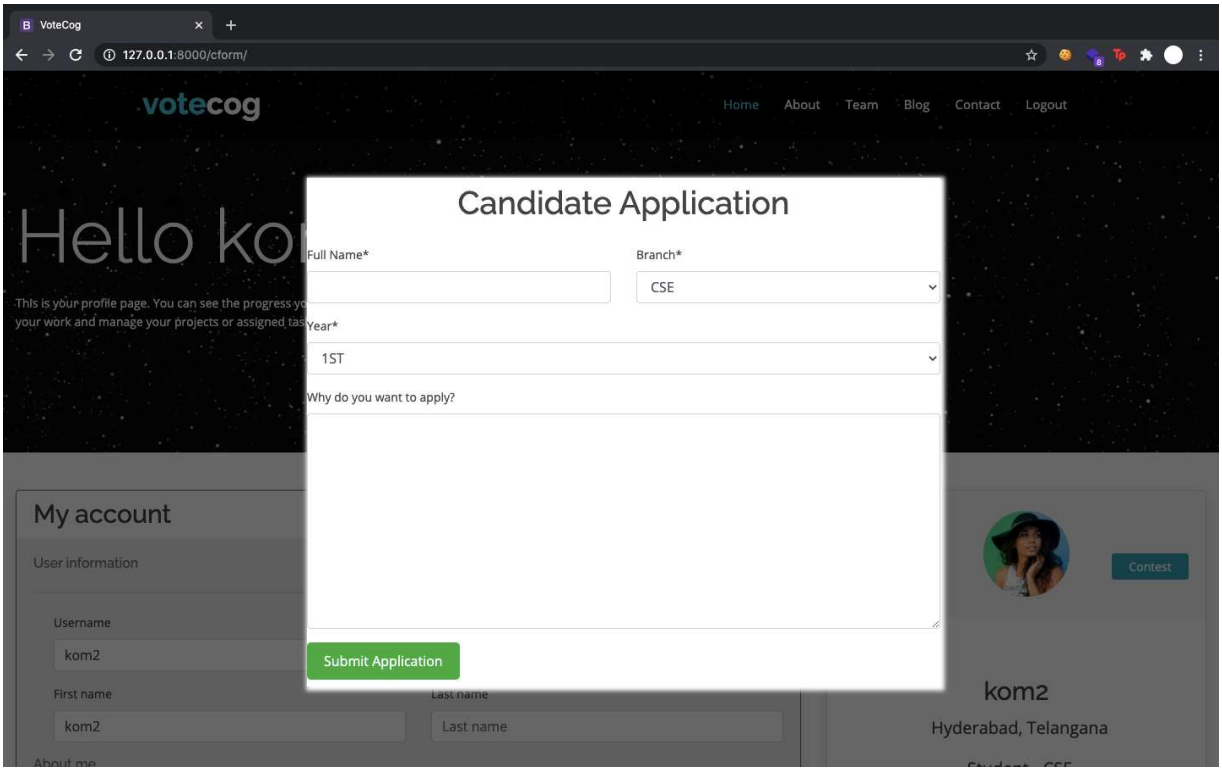


Figure 23 Candidate application form

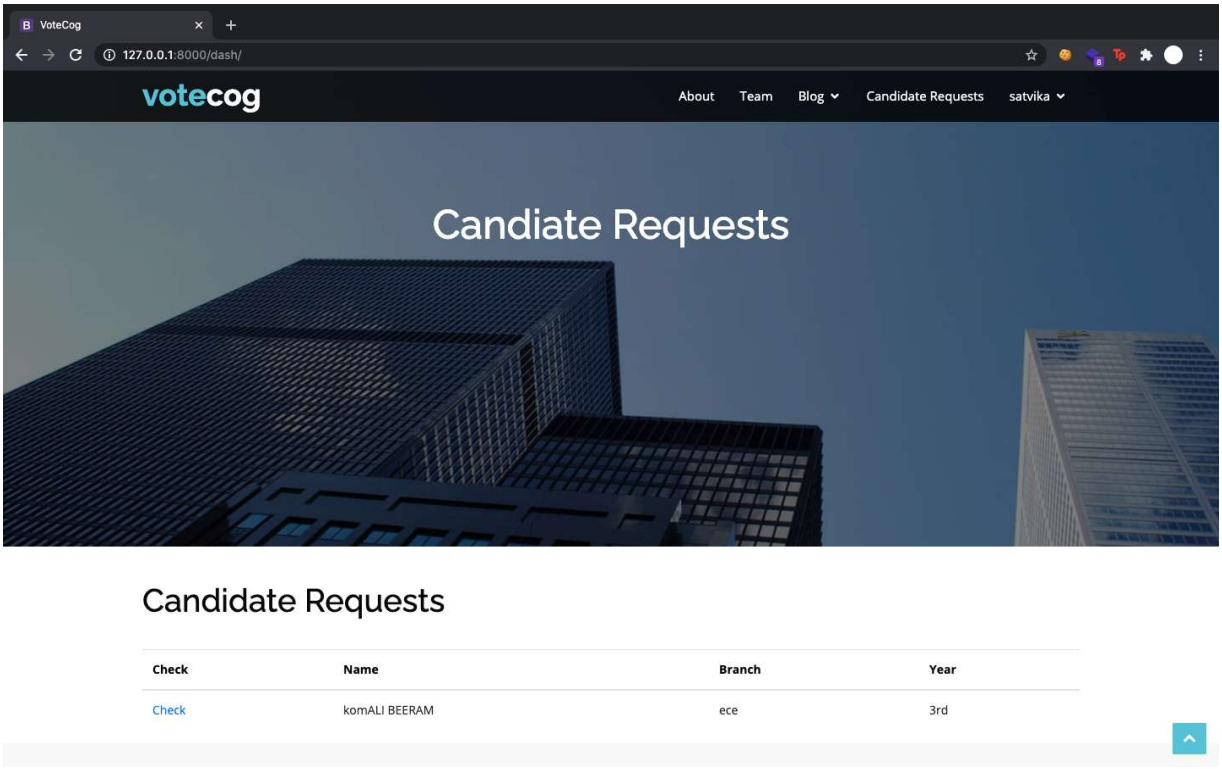


Figure 24 Candidates applications

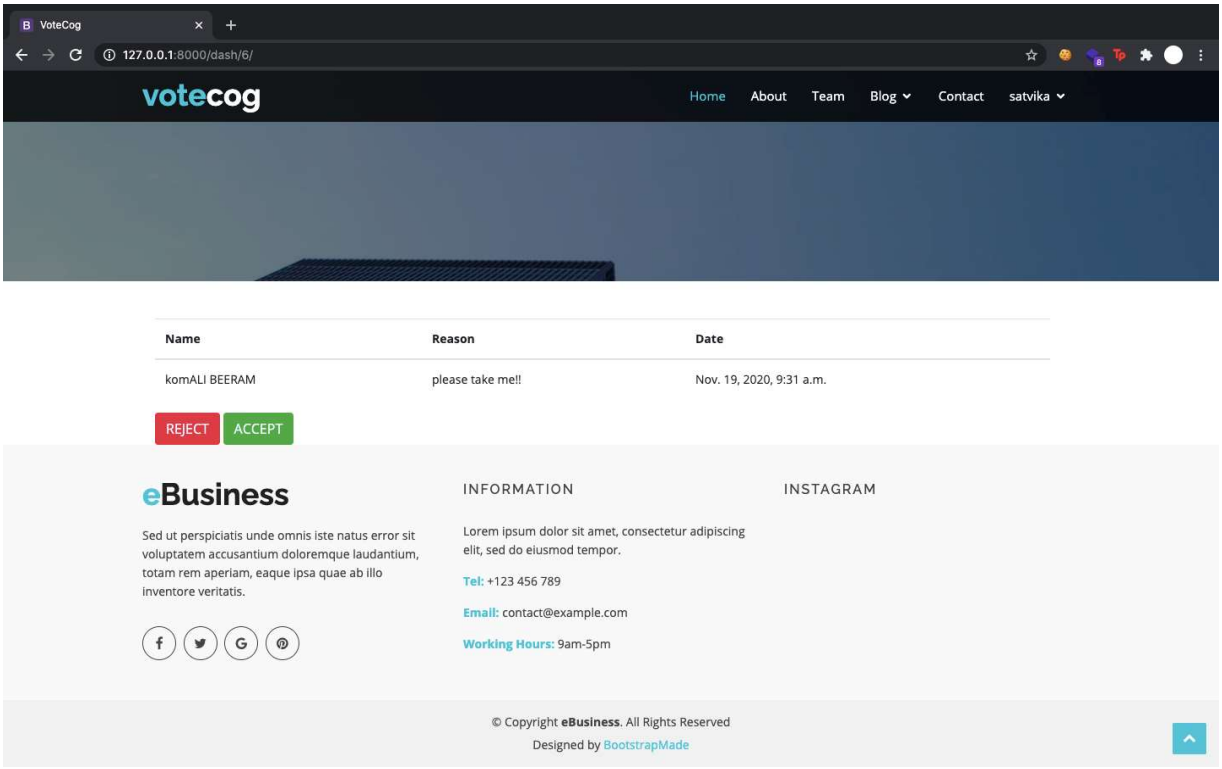


Figure 25 Accepting and rejecting candidate applications

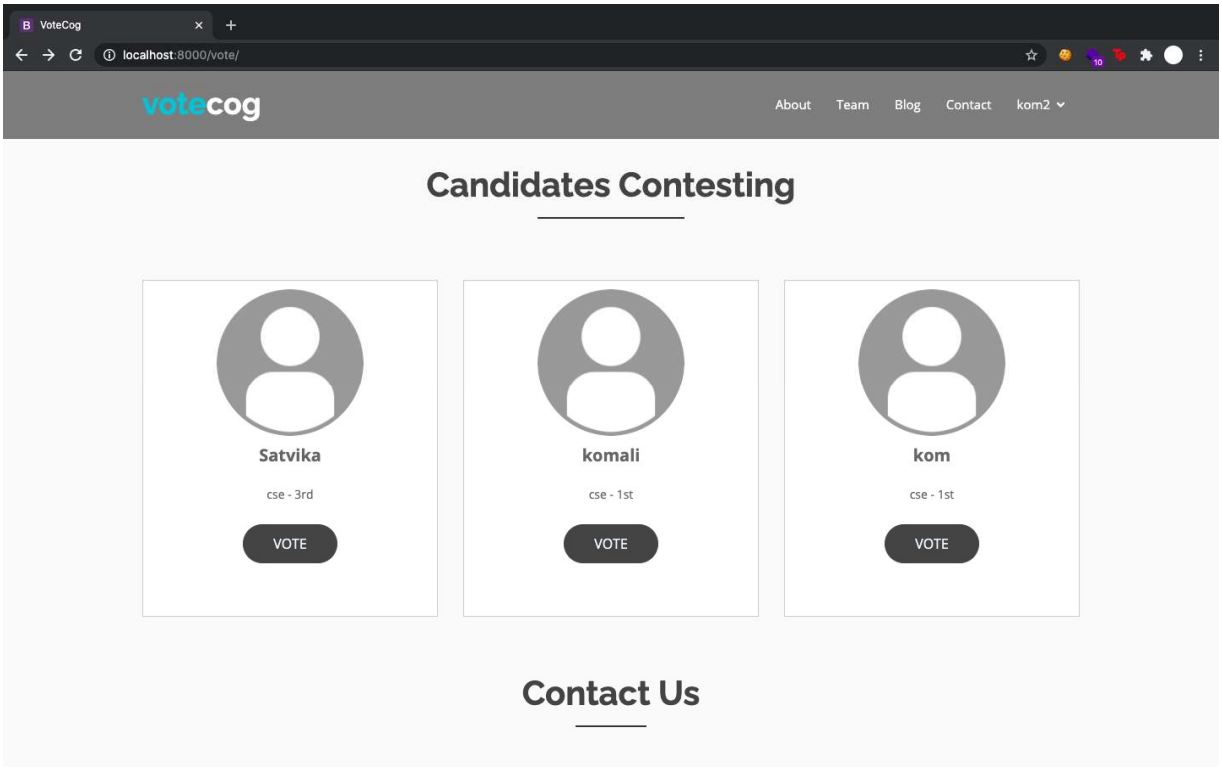


Figure 26 Voting portal

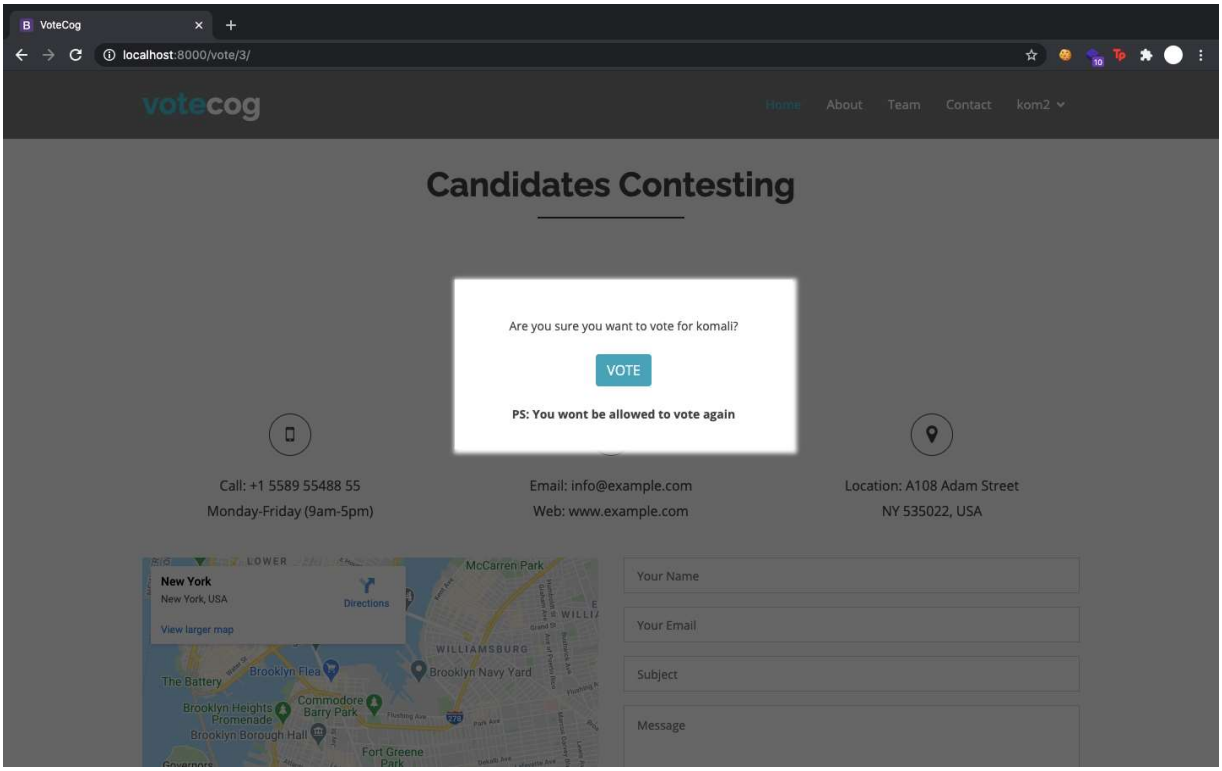


Figure 27 Vote confirmation

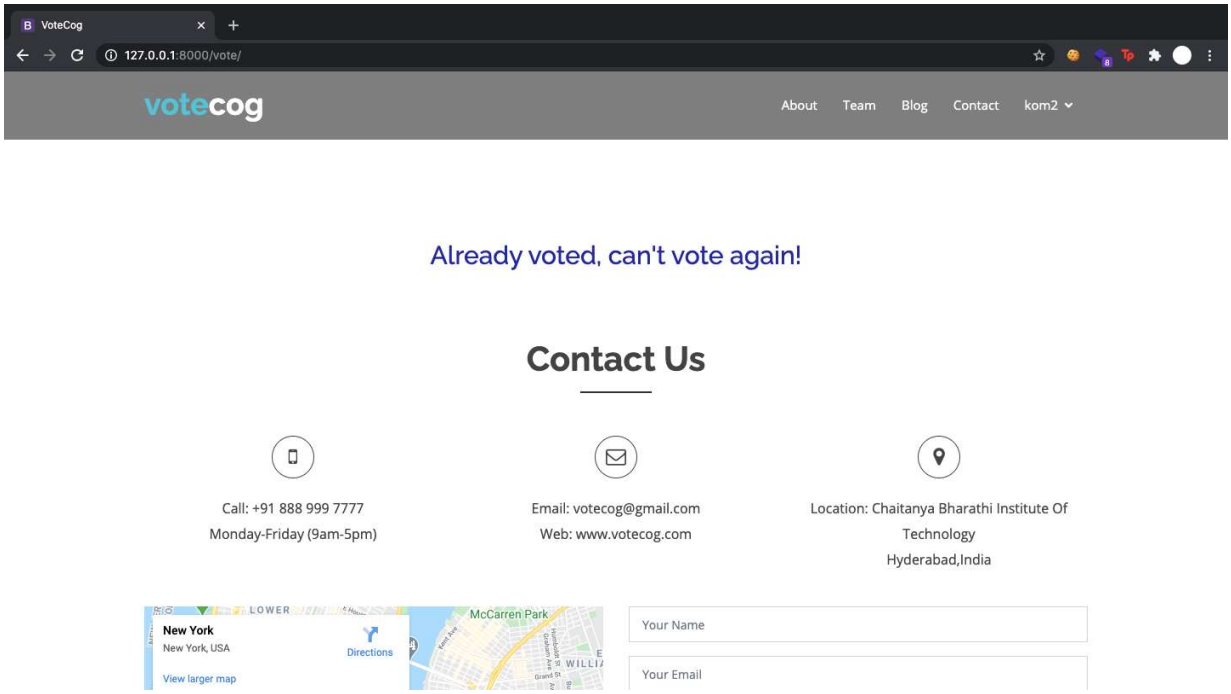


Figure 28 Attempt to vote again.

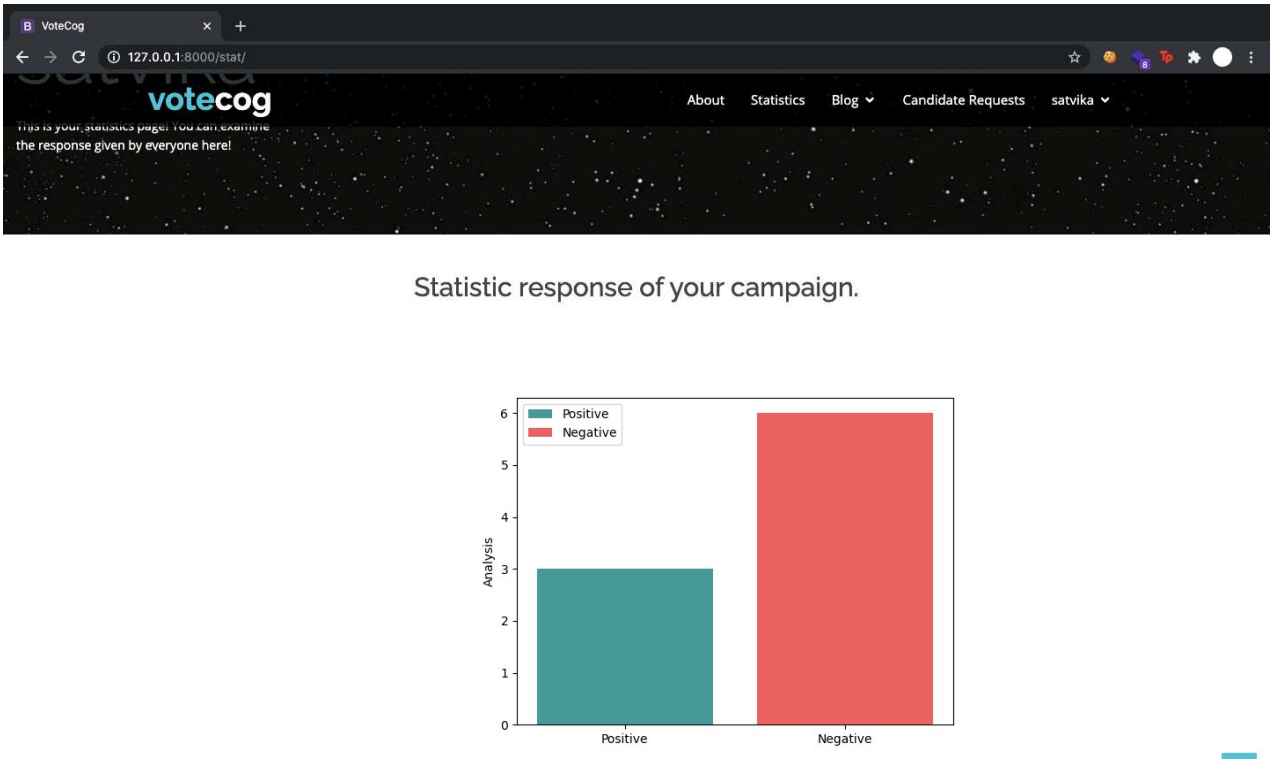


Figure 29 Campaign Analysis

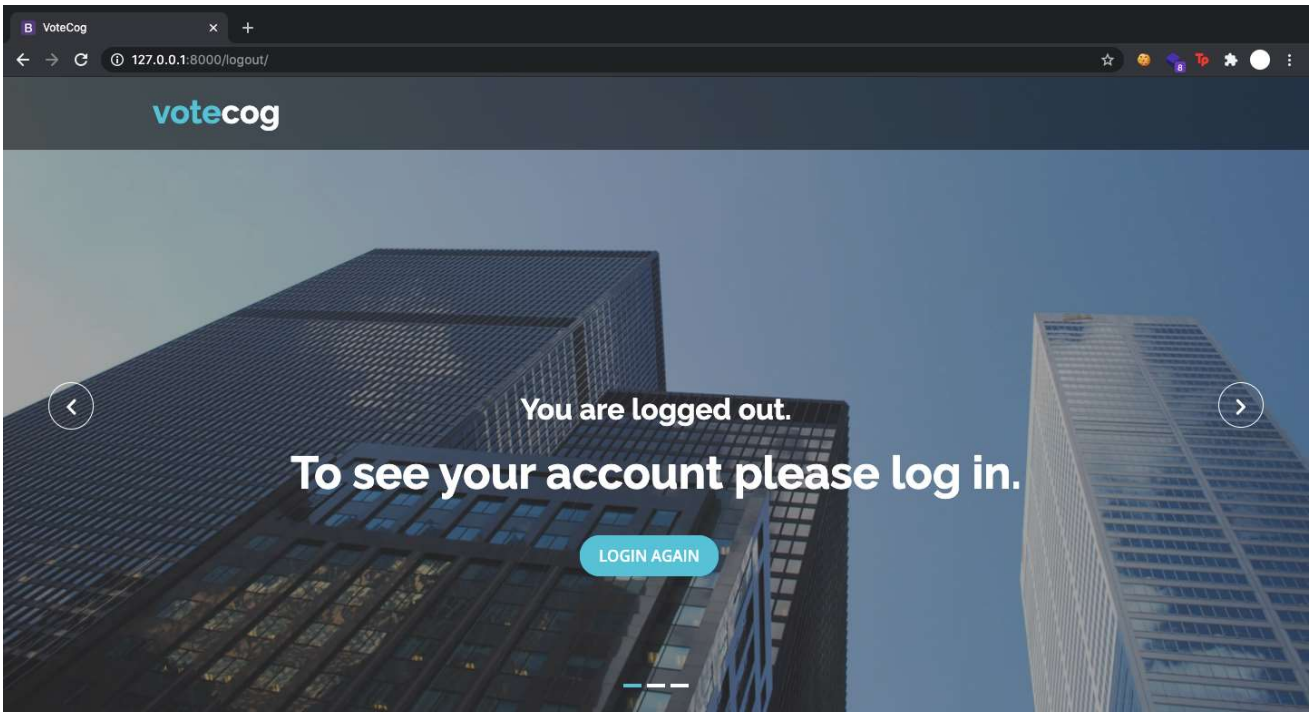


Figure 10 Logout page

CONCLUSION & DISCUSSION

CONCLUSION

In view of the current situation and not being able to hold voting locally, we decided to build this website to help security, increase in participation, accessibility and precision to the utmost. This website helps you maintain the anonymity and takes in your voice due to its reliability. Being CBIT Students and appreciating the great freedom our teachers provide us, we thought of using this on the Student body elections for our college as it is essential that all students get to decide their representative. In the whole we as a team discovered new technologies and learnt new things in the process which will help us vastly in our future. We hope that this prototype helps everyone in the real world as well.

FUTURE WORK

Below mentioned are the features we are hoping to add in the near future, in order to, make our system more accurate, secured and efficient.

- A machine learning model for the admin to see the statistical growth of each candidate.
- Ability to add custom pictures by the candidate with respect to his post to enhance the reach of his campaign
- Give admin the ability to open and close the voting portal as per the requirements of the student body.
- Add iris detection for students to login, so that other peers don't misuse the voting rights of a person. Thus, we plan on adding more security to our prototype.
- Give admin rights to withhold any candidate's candidature if he/she refrains to follow the etiquettes to be followed.

REFERENCES

- https://www.digitalocean.com/community/tutorials/how-to-perform-sentiment-analysis-in-python-3-using-the-natural-language-toolkit-nltk#:~:text=Sentiment%20analysis%20is%20a%20common,Python%2C%20to%20analyze%20textual%20data
- https://en.wikipedia.org/wiki/Sentiment_analysis#:~:text=Sentiment%20analysis%20(also%20known%20as,affective%20states%20and%20subjective%20information.
- https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17
- https://docs.djangoproject.com/en/3.1/
- https://pypi.org/project/django-bootstrap4/
- https://getbootstrap.com/

APPENDIX

Firstly, we made an outline of all the essential features that we wanted to include in our website. Making these layouts helped us easily get on with the essential database features we required. Initially we made a simple UI connecting to the entire backed and making sure that there is high security between accessing the pages with the rights a person has i.e. admin, candidate, voter.

Next, we collected data for the sentiment analysis using natural language processing. Here we used twitter samples and trained about 7000 records and tested for another batch of the same. We made sure we removed the noise from our data before feeding it to our Naive bayes classifier. This helped us get pretty 99.9 percent accuracy on our required data. Lastly, we made sure to integrate our backed with a much more aesthetic frontend and made sure our entire prototype worked perfectly for anyone to use.