HOW TO SETUP

1- **git clone https://github.com/flash1452/NotificationSystem.git**

2- INSTALLING NPM - **sudo apt-get install npm.**

3- INSTALLING NODEJS - **sudo apt-get update && sudo apt-get install nodejs**

3- Navigate to the cloned directory in the terminal.

4- **sudo npm install**

5- npm Modules to be installed -

  -- mysql

  -- fs

  -- body-parser

  -- crypto

  -- express-session

  -- path

  -- websocket

  -- http

NOTE- Use **sudo npm install your_module_name** to install any module mentioned above.

 6- Create a **new database named of your choice** in mysql. Import the **socialcops_db.sql in mysql** in that database and **change the username, password and database name in app.js** according to the system on which it is being run.

STEPS TO RUN-

1- Open two terminals. Navigate to cloned repository in both terminals.

2- Execute commands **node app.js** AND **node server.js** respectively in the two terminals.

3- Open **localhost:3000/home.html** in your browser.

4- Login with any username and password of your choice.

   NOTE - Two users cannot have same username.

5- After getting the **Login Successful** message, run the url **localhost:3000/sublist.html** in your browser to select the users from checkboxes you want to subscribe to.

6- Click on **Submit Subscription**.

7- If there is any change in current subscriptions from the previous subscriptions, then the **records in database gets updated** and **notifications are released** to all the users who are subscribed to the current loggedin user.

Tools/Components/Languages Used -

1- Nodejs
2- Node Websockets
3- Crypto Library
4- JS Websockets

WHY NODEJS -

1- NodeJs is a fast language when its comes to development.

2 - In order to successfully execute a project related to notifications, there has to be a socket connection between the server and the client and there has to be a valid connection between sockets used with JS and sockets used with Backend language. NodeJs is the most apt language to set a socket connection with the JS Websockets because of its library websockets.
Otherwise in other languages there is always a requirement of Handshaking between client and server which is quite costly and complex.

3- Crypto library was used to crypt the current loggedin userid so that its easier to authenticate a user and secure at the same time.

CODE DETAILS -

1- **User has to login before visiting its subscriber list or before changing his subscription.**

2- **Logged in user** details are stored in **table users.**

3- Login can be done in one step without registration first . If user has entered a new username which is not already there in the table user then his details are inserted and if he has used the username before then password is checked. If password matches only then he is allowed to login.

4- **Session varibales are set using userid, a hsh generated randomly for security reasons and username**. For every operation **hash is checked if it is set or not** in table **currently active users** corresponding to the correct userid and its **random so a user cannot set a hash** and then access the subcription.

5- **Entries in database** regarding users to whom the current loggedin user is subscribed to is done **only when the user has done some changes to its subscribers lis**t otherwise no entries or updation is made.

6- Notifications are generated by having a **mapping of userids that have loggedin** and the ip adress of those users and then **whenever there is a update or new entry in table submapping** then notifications are generated. **Notifications are also stored in notification table with default status as 1 meaning they are unread and automatically loaded whenever sublist.html is accessed.**

7- if a **user is not logged in** then his **notifications are stored in notification table** with status as 1. When a **user clicks on the notification symbol** in sublist.html then **all notifications related to that user are considered read** and their **status are set to 0** in notification table,.

8- **Notifications can be accessed by hovering over green bell** symbol and you can see a dropdown containing all notifications. As soon as user hovers over a bell all notifications with status as 1 are updated to status = 0 representing that user has read the notification.

Note - Every operation can be done only when user has logged in otherwise user will not get access.