



Dharmsinh Desai University, Nadiad

Faculty of Technology, Department of Computer Engineering

B.Tech. CE Semester – VI

Subject: System Design Practice

Project Title:

Go-On

Submitted By:

AkbariRajkumar CE005 17CEUOG126

BeladiyaDarshan CE013 17CEUBG057

KoratVivek CE063 17CEUOG128

Guided By:

Prof. Hari Om Pandya

Assi. Professor, CE Dept



Dharmsinh Desai University, Nadiad

Faculty of Technology, Department of Computer Engineering

CERTIFICATE

This is to certify that System Design Practice entitled “Go-On” is the bonafied report of work carried out by

1) AkbariRajkumar	CE005	17CEUOG126
2) BeladiyaDarshan	CE013	17CEUBG057
3) KoratVivek	CE063	17CEUOG128

Of Department of Computer Engineering, Semester VI, academic year 2019-2020,
under our supervision and guidance.

Guide

Prof. Hari Om Pandya
Assistant Professor of
Department of Computer
Engineering, Dharmsinh Desai
University, Nadiad.

HOD

Dr. C. K. Bhensdadia
Head of the Department of
Department of Computer
Engineering, Dharmsinh Desai
University, Nadiad.

Content

1. Abstract.....	4
2. Introduction.....	5
3. software requirementspecification.....	6
4.Design.....	7
4.1 Usecase diagram.....	7
4.2 Classdiagram.....	8
5. Implementation Details.....	9
6. Screen Shots.....	10
7. Conclusion.....	13
8. Limitation & futureImplementationDetails.....	14
8.1 Limitations.....	14
8.2 Future implementations	14
9. Bibliography.....	15

1. Abstract

A Solution for tracking the live things.

- what's going on in nearest places?

- current affairs as well as current videos of places.

- in shorts anyone can share their moments and anyone can see everyone's moments.

- this can be useful in current days also (corona-effect).

2. Introduction

-Go-On is one intermediary between users for sharing their moments as well as searching for moments .It is just storing live videos and searching in them with an efficient algorithms.

-searching is based on live coordinates.

Tools/Technologies

-Node-js

-RTCMultiConnection

-Ajax/jQuery/css/bootstrap

-socket.io

-HTML5 GeoLocation

3. Software Requirement Specifications

1)User

R.1. GoLive

Input : description about place, Location

Output : Live video running on server

R.2. ShowLiveVideo

Input : select live video to show

Output : Live video streaming

R.3.Fetch_Location

Input:it fetches location of users using HTML5Geolocation

R.4. Search

R.4.1. nearest_search

Output: based on location coordinates it fetches data from folder and shows in well defined format

R.4.2.recent_search

Output:based on recentness of videos (timestamp) sorts and shows on screen

R.5.Thumbnails

Output:shows All videos from folder with best

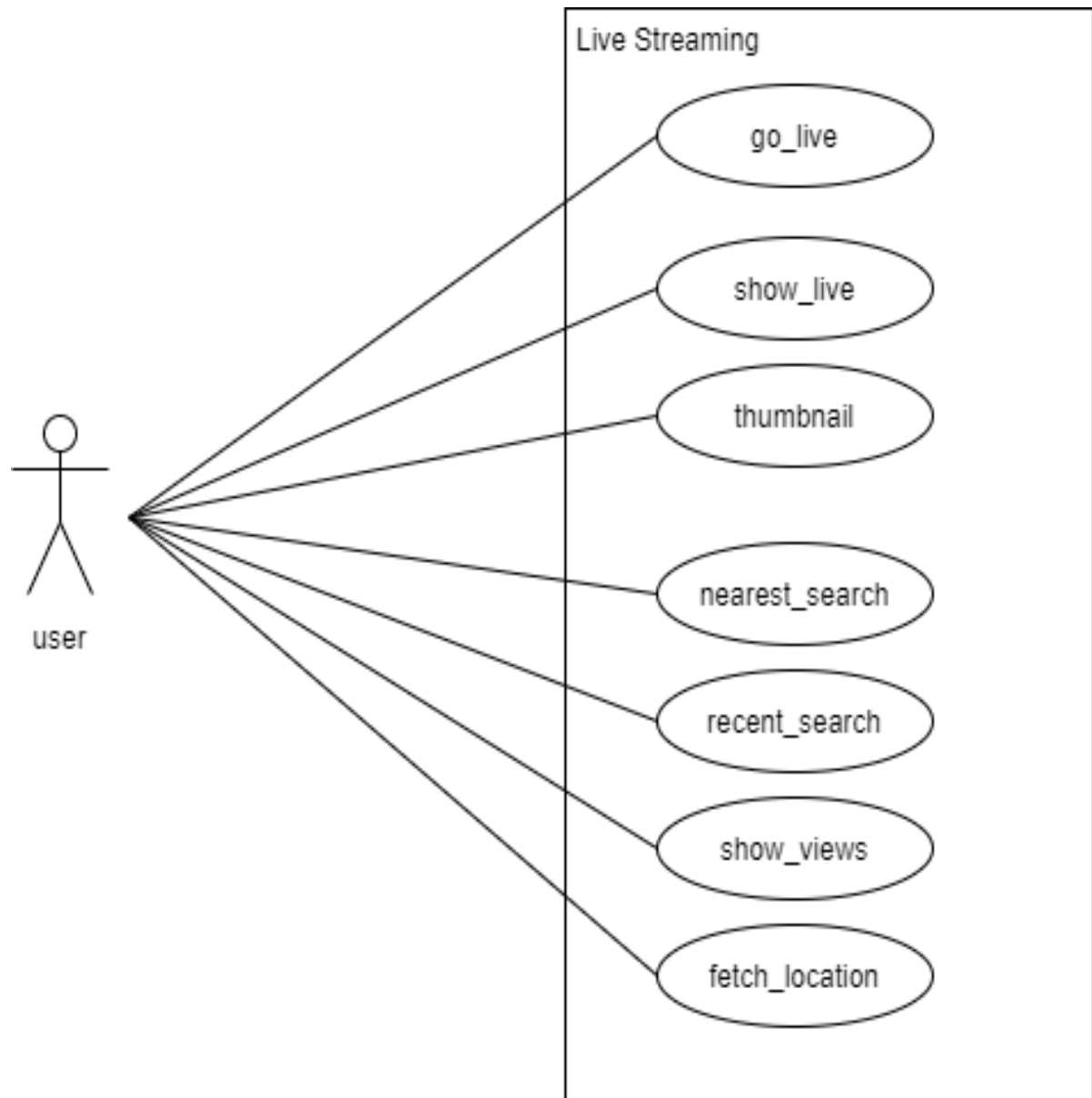
algorithm R.5.1.Search (inherited R.4)

R.6.show_views

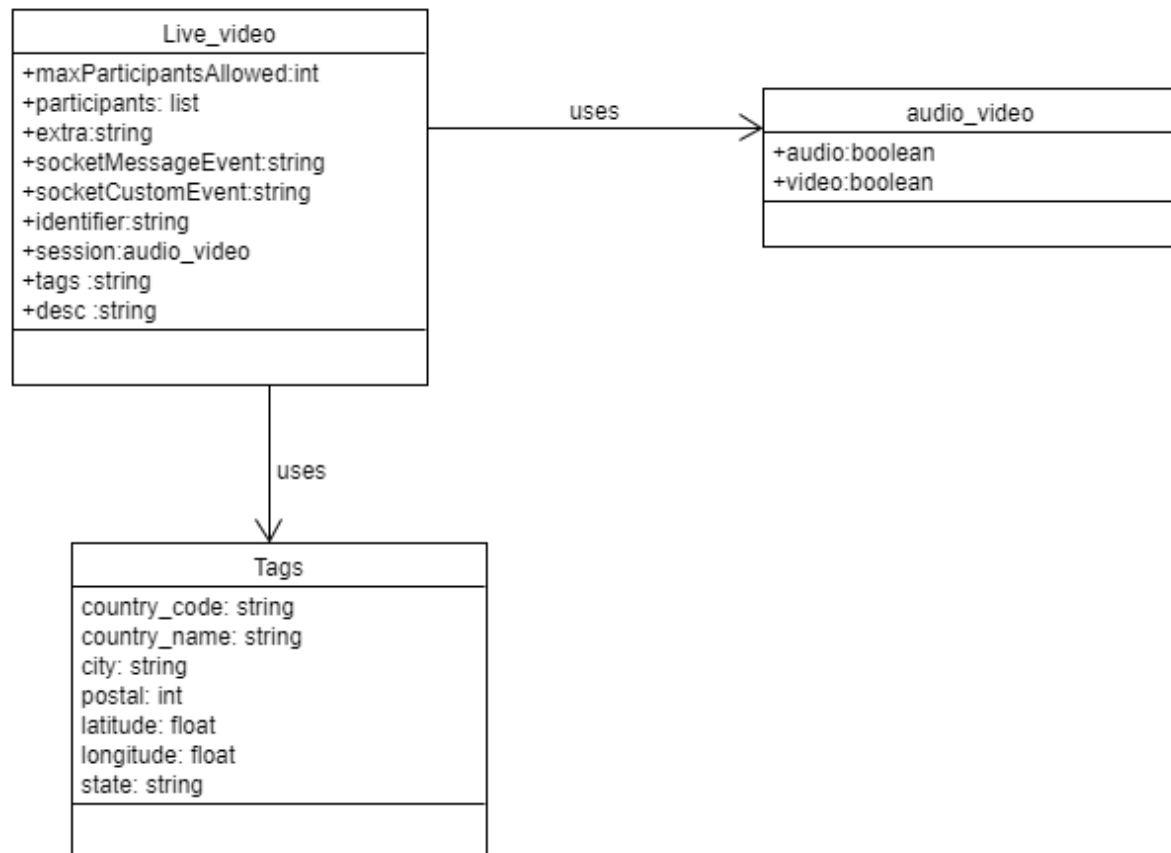
Output: Dynamic views on live videos

4. Design

4.1 Usecase_Diagram:



4.2 Class Diagram



5. Implementation Details

- Live Streaming:

- Live Streaming videos using RTCMulticonnnection library and socket .io

- storing all videos in our folder with their descriptions and live coordinates

- Live Location:

- live coordinates are fetched using HTML5 geolocation API

- Search:

- 75% efficient searching algorithm

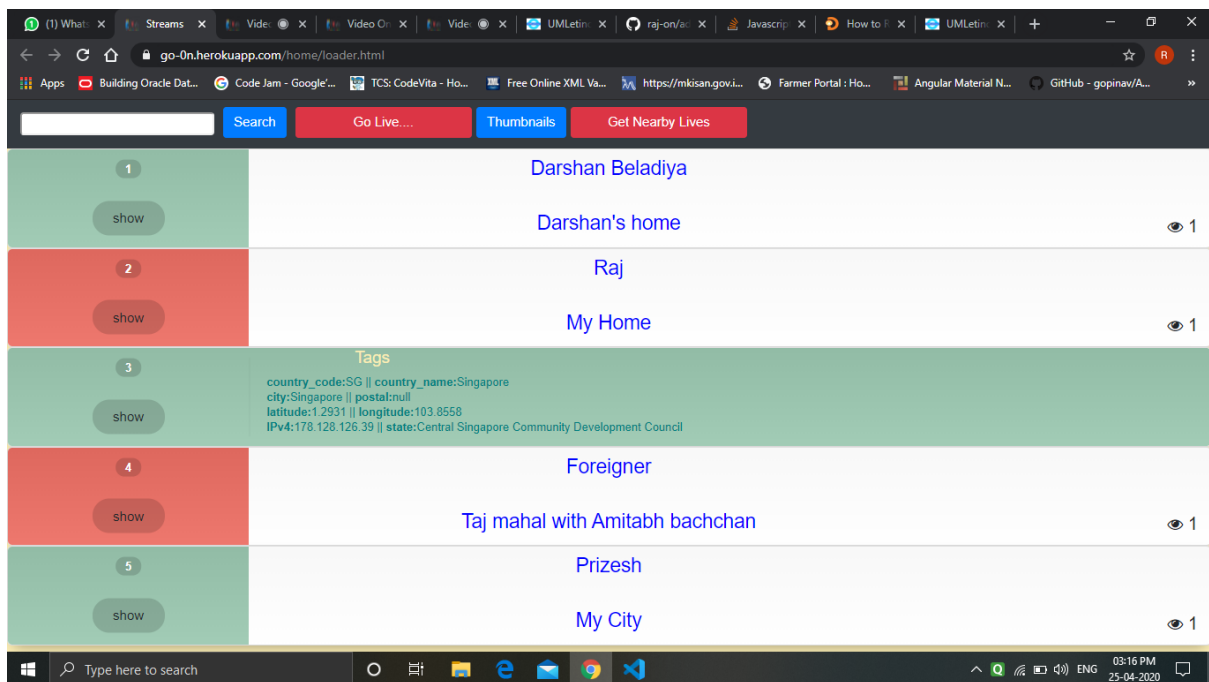
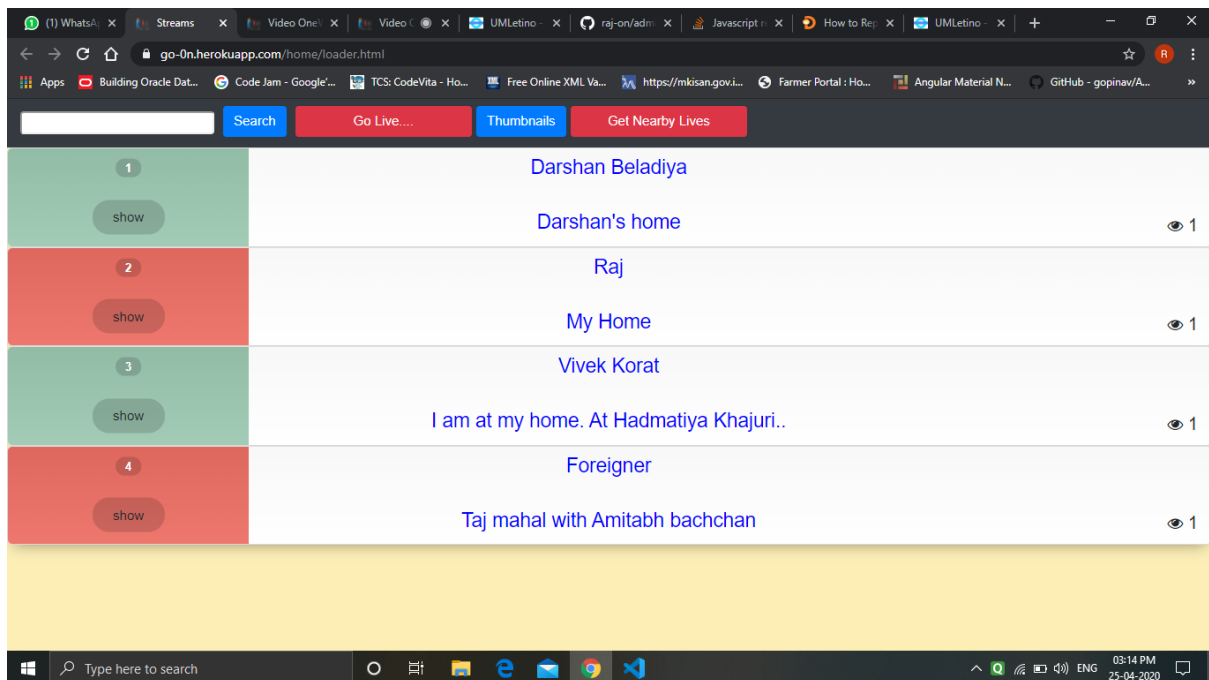
- matches using frequencies of characters

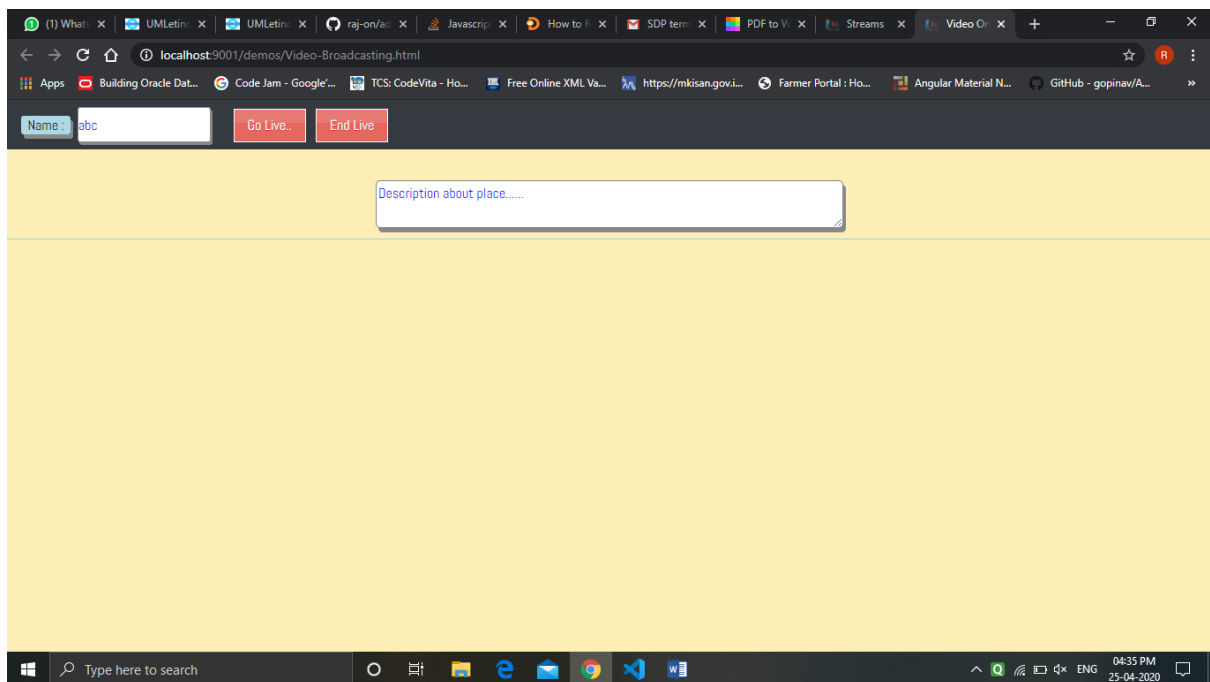
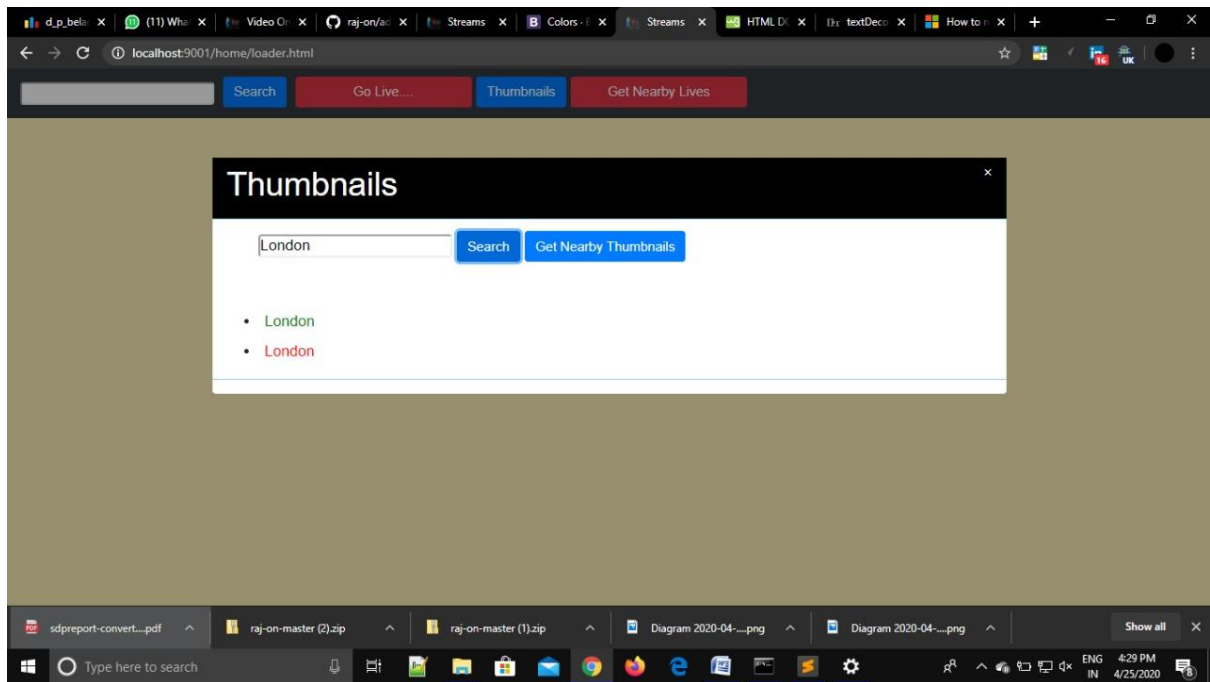
- nearest search: shows nearest videos using HTML5 Geolocation API and fetches location coordinates

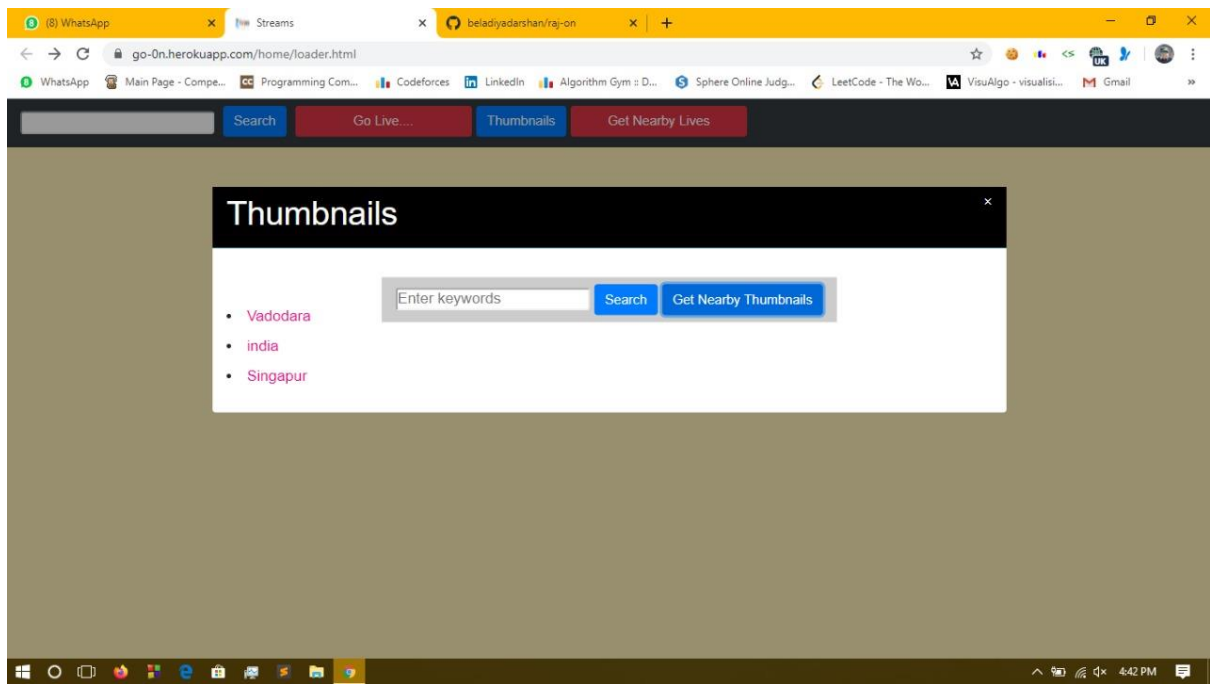
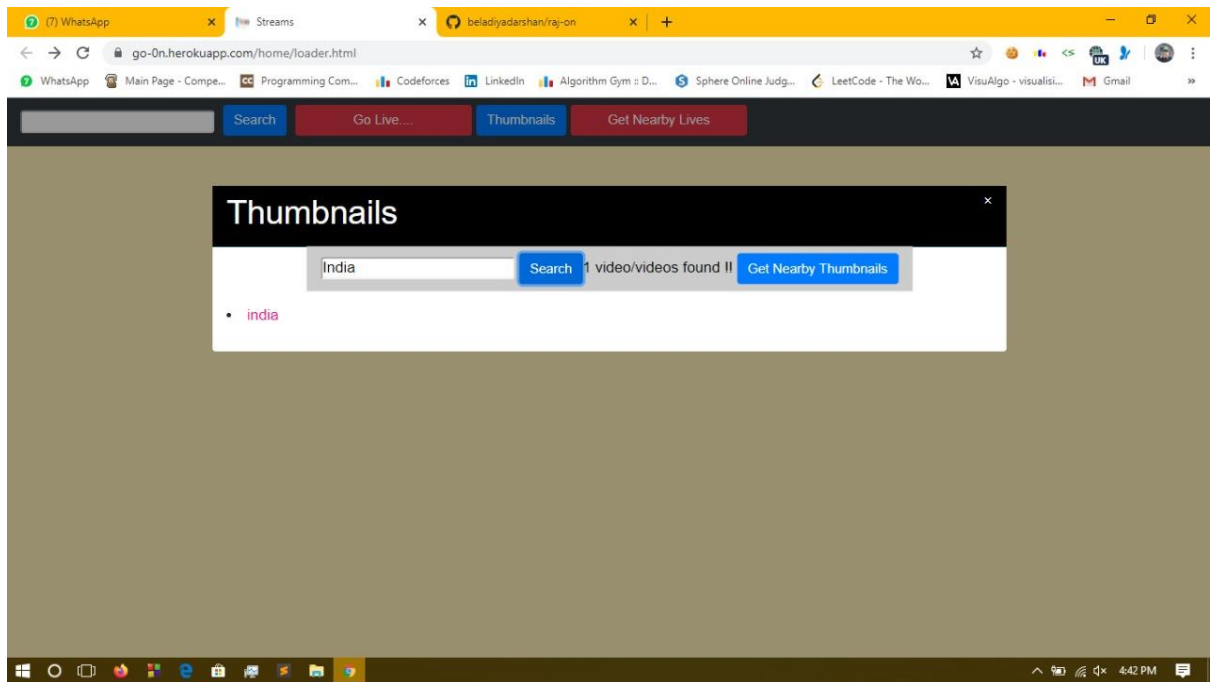
- LiveViews:

- It shows live views on live videos for references as well as searching purpose

6. screen shots







7. conclusion

-It is useful in critical situations as well as happy moments of day 2 day life.

-Everyone will have news of every moments of every places

-Also nearest new updating instantly so very useful

8. Limitations and Future Enhancements

8.1) Limitations

-Space limitations on server as well as limited live videos with server limit as well as storage limit.

8.2) Future Enhancements

-ML detections for videosfacial

-recognition

- effects like tiktoketc...

9. Reference / Bibliography

- ▯ Standard Documentation
<https://doc-kurento.readthedocs.io/en/6.13.0/tutorials/node/tutorial-magicmirror.html>

- ▯ Exmples
<https://github.com/muaz-khan/RTCMultiConnection>