Clipgur

Server

http://ec2-34-209-86-253.us-west-2.compute.amazonaws.com/

AWS Login

https://console.aws.amazon.com
info@clipgur.com
Testing123$

How to SSH

```
chmod 400 clip.pem
ssh -i "clip.pem" ubuntu@ec2-34-209-86-253.us-west-2.compute.amazonaws.com
```

Application

- Upload a video (YouTube URL or Local File)
- Select start time and end time
- Create a clip

MORE FUNCTIONAL, LESS PRETTY
GOAL IS TO DEMONSTRATE EXTENSIVE CAPABILITY WITH ANGULAR & LARAVEL

Project Overview

- Create basic login/logout/register views to accompany existing API (as simple as possible)
- Update API to support videos from local device
- Link created clips with user accounts (if logged in)
- Develop share to Instagram button on
- Create basic history view for logged in users
- Show proficiency with Material Design Directives usage

Frontend

- Use Angular Material Directives
- Take existing frontend on clipgur.com and transfer to new server
  - Ignore “Trending Videos”
  - Ignore “Search”
  - Ignore Static Pages (/about, /terms, /policy, /contact)
  - What to include
    - Header

    - Media upload part (YT link OR Upload Video from PC)
    - Media view page
    - History (new)
    - Login/Logout/Register buttons & views (new)
- Add basic login/signup (API is ready for this)
- Add logged in user state in header
- Add logout function in header
- No email activation required
- For logged in users create a “History” page
  - Show all videos that logged in user created

Backend

- Create DB model for videos
  - id
  - user_id
    - applies only if created by logged in user
    - otherwise “Anonymous”
  - type
    - youtube/upload
  - title
  - start
  - src
  - duration
  - yt_source
  - created_at
- Update API with new endpoints to support following logic
  - Allow uploading videos from computer (via file [mp4, webm, avi])
  - Via YouTube link (exists in a crude form now in HTTP controller but can be much improved)
  - Clip videos (exists already but repurpose for file uploads)
    - inputs are start & duration
    - ffmpeg required on local machine
  - Save clipped videos to DB
    - src is directly on server
  - Upload clipped videos to Amazon AWS
    - Once uploaded and on Amazon updated src on DB
- Share to Instagram Button on video view page (https://github.com/LookHin/instagram-photo-video-upload-api)