# MVSR Engineering College, Hyderabad - 501 510
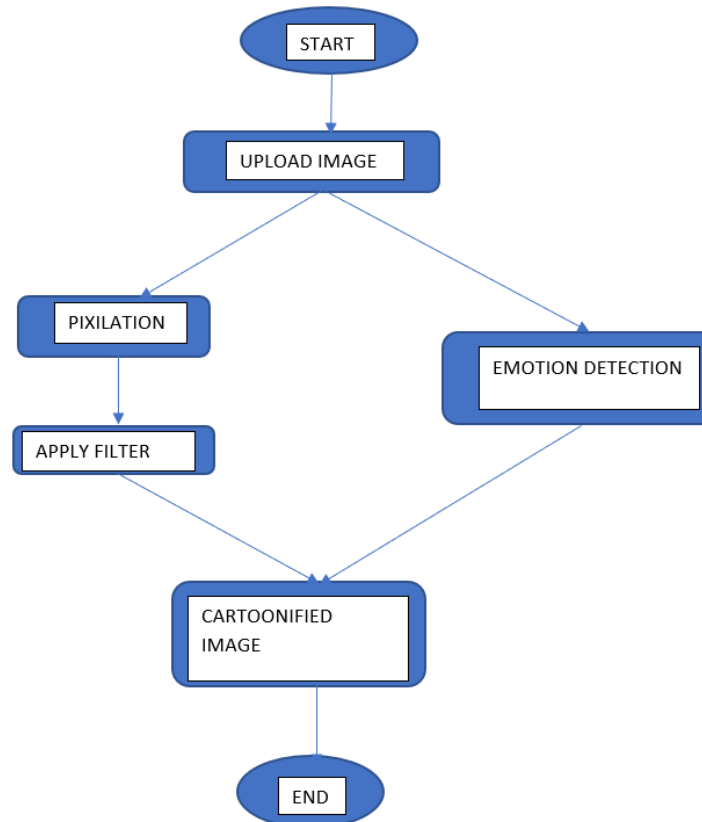
# Department of Computer Science and Engineering

**Project Name: Cartoonifying an Image**

**Objective:** In a world that runs by means of communication, it is important for anyone to visually express their feelings, even virtually. Emoticons and stickers can be used to communicate, but they do not convey our expressions. Many people still feel insecure to share their images, as images of individuals have been misused in few cases. In this situation, a growing need for an application, that can eliminate the insecurities and help people convey their expressions arises. It is also important for a machine to automatically detect the emotion that a person feels by looking at their image. The machine can help lighten up the mood of a person if they are feeling sad or depressed and are alone. The Emotion Detection tool helps to recognize the feeling of a person.

```
                          ( START )
                              |
                              v
                      [ UPLOAD IMAGE ]
                        /           \
                       v             v
               [ PIXILATION ]   [ EMOTION DETECTION ]
                     |               |
                     v               |
              [ APPLY FILTER ]       |
                      \             /
                       v           v
                    [ CARTOONIFIED
                        IMAGE ]
                          |
                          v
                       ( END )
```

## Implementation of Module-1

**Software Environment Used** : Jupiter Notebook (or) Google Colab, Flask, AWS
**Major Functions used** : Easygui, OpenCV, Matplotlib, imageai
**Number of lines of code** : 65

Test case 1:



Original Image        FILTER = 2        Grayscale Image

Test case 2:



Original Image        FILTER = 5        UV Filter

Marks:3

Implementation of Module-2

**Software Environment Used** : Jupiter Notebook (or) Google Colab, Flask
**Major Functions used** : CNN, Pandas, Keras
**Number of lines of code** : 30

Test case 1:



→ You look so happy, keep going. Have a good day!

Test Case 2:



→ You look sad, cheer up and watch a movie.

Marks:3

Integration & Deployment

# Integration

Developed 2 different algorithms for cartoonifying an image and emotion detection. These 2 algorithms have been integrated into a single web application using Flask.

We have built a machine learning model using CNN to detect the emotion of face in the image and deployed it, which can be retrieved using an HTML web page.

# Technical challenges facing

• Giving only specific users access to reverse engineer the process.

• Dataset with improper attributes

• Selection of Appropriate Filters

## Project Demo

Demo of project

# Cartoofying an Image

Drag & Drop Image

Choose File  No file chosen

**Select a filter:** Gray Scale  ⌄          **Cartoonify**

Marks:3

Results

## Objective Evaluation

Accuracy of the Machine Learning Model
Magnitude of Cartoonification

We have used a dataset named fer-2013.csv which contains the images needed to train the Machine Learning Model and the emotions corresponding to the particular image.
After training the model using CNN, we have obtained an accuracy of 69%.

Marks:2

Conclusion & Future scope

## Conclusion

We have built an application which performs the dual functioning of cartoonifying an image, as well as detecting the emotion in the image.

Both the functions give accurate results and this application proves to be helpful among various domains, i.e. , to post the images on social media, or share images online without security and privacy concerns, expression detection among the individuals.

## Future Scope

Currently, we need to take a picture of the person to apply the filter, but this can be improvised by adding the live feed of the camera, so that the filters can be compared without capturing a picture.

We can also include and apply various filters and experiment with various colour combinations.

The present Machine Learning Model is trained only to detect six standard emotions. This can be improvised by training the model with new set of emotions. We can further improve the application such that the emotion can also be detected in the camera live feed.

Marks:2