# FaceFilter APP

MITIN SHARMA

## INTRODUCTION:

This is a simple standalone chat application, wherein, a user can communicate with the other users around without even knowing their phone number or email.
It is a very simple to use application, as the user just needs to set a username and choose an avatar, and they're ready to SWAP IT.

## HOW IT WORKS?

As soon as the user enters a username and chooses an avatar, he is directed to a new activity. This activity contains the list of devices that have the app installed and that are connected to the router.  You can select the user based on the username provided and can directly initiate a chat. If there are more than two devices connected to the network, you can even start a new group chat together.

**ADVANTAGES:**

a) It is simple and easy to use.

b) Doesn’t require any login credentials. It’s pretty much open.

c) Can even be used to share files.

## MORE ABOUT THIS:

Our Application is designed to support only Android devices.

**Technical: -**The mobile application will be developed in Android so, it requires the mobile device to run on Android OS. The communication between these devices is done using Wi-Fi Direct , so it is essential that the devices have Wi-Fi Direct support.

## SECURITY:

In a chat based application, it is important to consider the privacy of the user as well. For this reason, we have used AES encryption to secure the messages. This is done using a 128-bit key.

```java
import java.security.MessageDigest;
import java.util.Arrays;
import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;
import javax.crypto.spec.SecretKeySpec;
import javax.crypto.spec.IvParameterSpec;

import javax.crypto.Cipher;
import javax.crypto.spec.IvParameterSpec;
import javax.crypto.spec.SecretKeySpec;


public class Main {
    static String IV = "AAAAAAAAAAAAAAAA";
    static String plaintext = "Test Chat MSG\0\0\0"; /*Note null padding*/
    static String encryptionKey = "4582134745mitvis";
    public static void main(String [] args) {
        try {

            System.out.println("plain:   " + plaintext);

            byte[] cipher = encrypt(plaintext, encryptionKey);

            System.out.print("cipher:  ");
            for (int i=0; i<cipher.length; i++)
                System.out.print(new Integer(cipher[i])+" ");
            System.out.println("");

            String decrypted = decrypt(cipher, encryptionKey);

            System.out.println("decrypt: " + decrypted);

        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    public static byte[] encrypt(String plainText, String encryptionKey) throws Exception {
        Cipher cipher = Cipher.getInstance("AES/CBC/NoPadding", "SunJCE");
        SecretKeySpec key = new SecretKeySpec(encryptionKey.getBytes("UTF-8"), "AES");
        cipher.init(Cipher.ENCRYPT_MODE, key,new IvParameterSpec(IV.getBytes("UTF-8")));
        return cipher.doFinal(plainText.getBytes("UTF-8"));
    }

    public static String decrypt(byte[] cipherText, String encryptionKey) throws Exception{
        Cipher cipher = Cipher.getInstance("AES/CBC/NoPadding", "SunJCE");
        SecretKeySpec key = new SecretKeySpec(encryptionKey.getBytes("UTF-8"), "AES");
        cipher.init(Cipher.DECRYPT_MODE, key,new IvParameterSpec(IV.getBytes("UTF-8")));
        return new String(cipher.doFinal(cipherText),"UTF-8");
    }
}
```

Encryption key : 4582134745mitvis

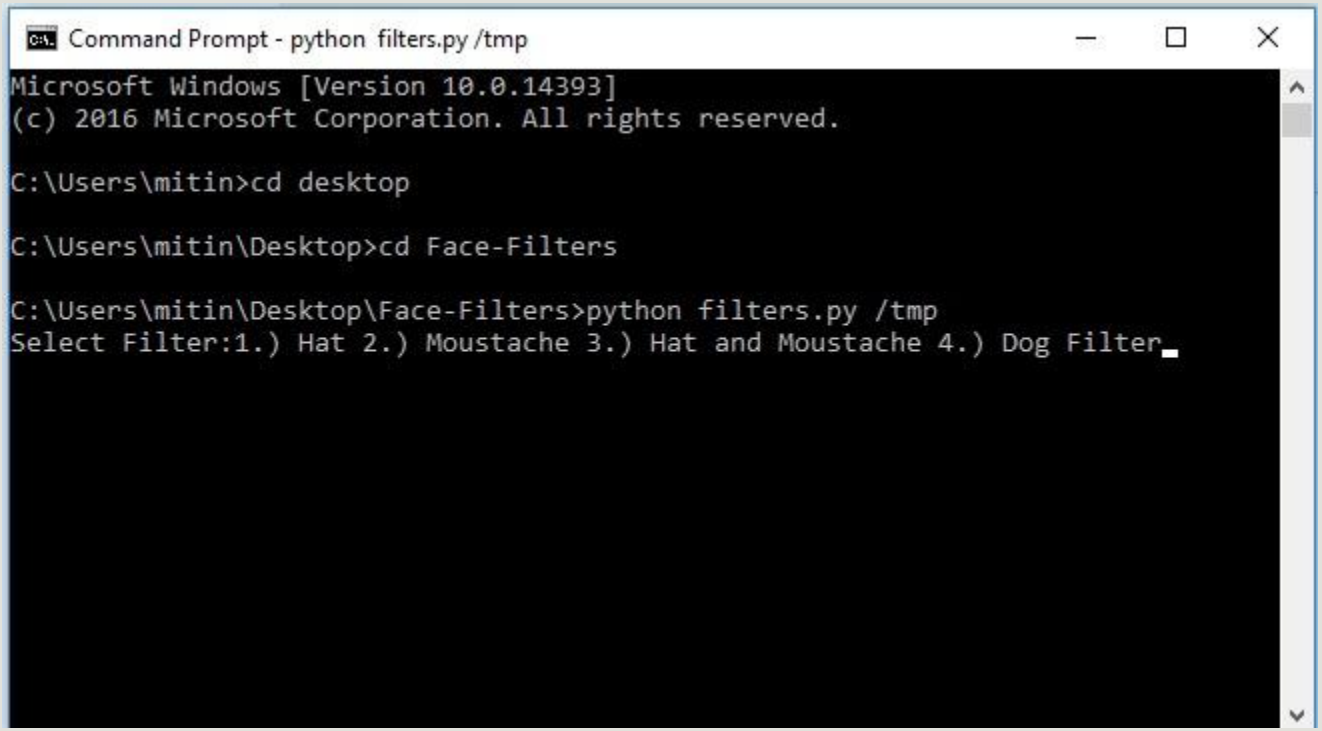
plain:  Test Chat MSG

cipher:  -116 -81 30 108 3 47 -101 73 58 -51 25 36 -110 -104 88 123

decrypt: Test Chat MSG

# SCREENS:

## HOME SCREEN:-


Command Prompt - python filters.py /tmp
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.
C:\Users\mitin>cd desktop
C:\Users\mitin\Desktop>cd Face-Filters
C:\Users\mitin\Desktop\Face-Filters>python filters.py /tmp
Select Filter:1.) Hat 2.) Moustache 3.) Hat and Moustache 4.) Dog Filter

**Hat Filter:-**

## Moustache Filter:-

## Hat and Moustache Filter:-

**Dog Filter:-**

# THANK YOU