

INDIAN INSTITUTE OF TECHNOLOGY KANPUR



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COMPUTER NETWORKS- CS425A

---

# Design of Application

COLLABORATIVE PAINTING

---

*Project Members(CN-08):*

AHSAN BARKATI(160055)

NITHISH BHUMIREDDY(160199)

SHIVANK GARG(160658)

SHUBH GUPTA(160668)

*Supervisor:*

PROFESSOR DHEERAJ SANGHI

# Contents

<b>1</b>	<b>Acknowledgement</b>	<b>2</b>
<b>2</b>	<b>Introduction</b>	<b>2</b>
<b>3</b>	<b>Goals</b>	<b>2</b>
<b>4</b>	<b>Features</b>	<b>2</b>
<b>5</b>	<b>Overview</b>	<b>3</b>
5.1	Modules Libraries Used . . . . .	3
<b>6</b>	<b>Problems Faced and their Solution</b>	<b>3</b>
6.1	Data formatting Errors with TCP . . . . .	3
6.2	Errors due to different resolution screens for different clients . . . . .	4
6.3	Issues related to voice integration . . . . .	4
6.4	Persistent Drawing related error . . . . .	4
<b>7</b>	<b>Future Tasks</b>	<b>4</b>

# 1 Acknowledgement

We express our gratitude to Professor Dheeraj Sanghi for providing us the opportunity to work on this project as part of this course. This project has given us the great opportunity to learn the concepts of socket programming, network and all round development related to development process.

Without his moral and technical support, we would not have been able to complete this effortful task. Overall, we thank him for providing us with the opportunity to make something creative along with gaining.

# 2 Introduction

Collaborative art is a form of artwork that involves working as team to create artwork, and each person's contribution is significant in a way to enhance creativity in artwork. Collaborative Painting application is one such platform designed by our group (CN-08) which let two or more painters, however apart, to paint on a single canvas. It let artists to draw simultaneously and that too with different color in various sizes brushes. If you feel unsatisfied with some part of drawing, don't worry, there is an eraser to remove it.

# 3 Goals

Goal of this application is to let different users paint simultaneously and they should have choice of eraser, colors and sizes. The application should not be restricted to particular resolution screen, but allow users of all screen sizes to paint without any limitation. Users should also be able to communicate to each other what they wish to draw on canvas. They should save the canvas and also have ability to draw over images.

# 4 Features

Our application has following features-

1. Different number of client can share the same canvas to draw simultaneously.
2. Users have choices of color and brush sizes.
3. Canvas is adjustable according to different screen sizes for various users and different resolution screen

4. Voice chat implementation to share plans and thoughts.
5. Easy to use scripts with scalability of as many clients one wish

## 5 Overview

We have used Python3 and its various modules(like tkinter, pyaudio, json, etc) to create this collaborative paint application. Client sends an event to server which broadcasts the information to every other client. An event consists of coordinates and color and brush size. So, every other client uses that information(event), to draw it on their canvas. We also run another server to implement voice chat. We used PyAudio to record the microphone stream which will be sent to voice chat server and it will broadcast the stream to all the connected clients. We used the tkinter canvas to make that possible. Tkinter color chooser allows us to choose the different combination of RGB for drawing.

### 5.1 Modules Libraries Used

1. Use of Socket programming to create TCP/IP sockets
2. Use of Threading to handle multiple clients
3. Use of Tkinter in Python3 to implement GUI
4. Use of JSON to format the data for sending and receiving
5. Use of PyAudio for voice chat
6. Other modules like select, sys, os and time are also used

## 6 Problems Faced and their Solution

### 6.1 Data formatting Errors with TCP

#### **Data was not structured properly**

Using JSON to format the data provided very structured way to send the data, but due to latency in data, the JSON strings get appended one after the other and they were not parsed properly. So we have to separate the JSON with '—' to ensure they are sent and received correctly.

## **6.2 Errors due to different resolution screens for different clients**

### **Coordinates(Pixels) vary for all different screen**

Absolute coordinates of screen were converted to relative coordinates. This lead to adjustment according to different screen.

## **6.3 Issues related to voice integration**

### **Some issues were faced during voice integration using PyAudio**

Issues were related to drivers and Microphone, which can resolved by checking the microphone is on using 'alsamixer' or 'pavucontrol'.

Also their were issues related to threading which was giving errors, these can resolved eventually.

## **6.4 Persistent Drawing related error**

### **Persistent drawing leads to error**

We thought of saving the tkinter drawing data to a log file which can be used by a new client to get updated canvas which other clients have drawn. But, we faced some issues on JSON data and other modules in implementing that idea. Due to this, persistent drawing idea could not be used.

# **7 Future Tasks**

1. **Implementation of Project Management** - Each client can work on a variety of other projects.
2. **Implementation of user login/signup** - Each project will require a unique key which will given by one who created the project.
3. **Introduction of various other paint tools** - Paint tools like shapes, fill color, and crop can implemented to give the artist more freedom.
4. **Creation of Checkpoints** - Various checkpoints can be created by clients so that they can revert back to saved canvas in case of any mistakes.
5. **Integration of scripts** - Currently, the audio scripts are not integrated in the main paint.py, which we wish to integrate with paint.py and add buttons related to audio in canvas.