SCRIBBLE IO

Summer Training Report submitted in partial fulfillment of the requirement for the degree of

B.Tech.

In

Computer Science Engineering

Training Coordinator

Ms. Nikita

By

AKSHAT GUPTA
50120802720

Bhagwan Parshuram Institute of Technology PSP-4, Sec-17, Rohini, Delhi-89

January 2023

INTRODUCTION

"Real Time Collaborative Whiteboard (Scribble IO)" is a simple web application that simply helps people to collaborate on various ideas and sketch it out. The users may be Students discussing any study topic, Teachers explaining students through whiteboarding and Professionals sharing their ideas while brainstorming. This whiteboard helps each and every user.

- PROJECT IDEA

  The idea behind my project is very nice and. I thought that I should make something that is "useful" for everyone in some way and also "easy to use", so the idea for this collaborative whiteboard came up. So far, I have implemented some of the useful features that will be handy to use for users.

- WHY I CHOSE THIS TOPIC

  I chose this whiteboard web application as my topic because I wanted to make a product that is "unique" and "helpful" for the people. Also, the other reason is that I am excited and motivated about making this web app and I like this kind of products and so this helped me to complete the project so far.

- PURPOSE OF THIS PROJECT

  The Main purpose of this web application is to provide ease to users while they are discussing any important or casual topic. And also, to provide easy access to users by directly providing the service through web, means there are various whiteboard websites out there but they require account but, in our case, the user can directly send the website link to the people that want to collaborate, discuss anything or sketch together.

- NEED OF THIS PROJECT

  - If the user is in hurry and don't want to create a new account on web for using whiteboard type of services then my web application fulfils this need.
  - The web application has the download option as well so that the user can save his/her work or sketch or any brainstorming idea on their Local computer.
  - Other websites don't allow you to download the board until you login, but in my case, the user can easily download without any login required.

SOFTWARE REQUIREMENT SPECIFICATION

- INTRODUCTION:

This document aims at defining overall software requirement for Online Collaborative Whiteboard. Efforts have been made to define the requirements exhaustively and accurately. The final product will be having only features/functionalities mentioned in this document and assumptions for any additional functionality/feature should not be made by any of the parties involved in developing/testing/implementing /using this product.

Purpose:
This specification document describes the capabilities that will be provided by the Online Collaborative Whiteboard. It also states the various constraints by which the system will abide. The intended audience for this document is the development team, testing team and end users of the product.

Scope:
The application will enable participants to interact with each other through means of graphics drawn on whiteboard. This will allow users to conduct teaching activities/playing activities (like scribble game). The application will greatly influence the way people collaborate.

- OVERALL DESCRIPTION:

The application will enable users to express themselves to other users using graphics/drawings. The application can be used to conduct teaching activites or playing games on the go.

- **User Needs**

The proposed system has the following requirements:

  - System needs to allow users to create custom room.
  - System needs to show whiteboard to every user in real time.
  - System needs to allow every user to interact/write on whiteboard in real time.
  - System must contain some method to change brush colour.
  - System must have some method to erase incoorectly drawn picture.

- **Assumptions and Dependencies:**
    - No user will sabotage the activity being conducted on whiteboard.
    - The users have sufficient knowledge of computers.
    - The users must know English language, as the user interface will be provided in English.

- **Functional and Non-Functional Requirements:**
    - **Functional Requirements**
        1. Create room: User must be able to create a custom room to invite other users in that room.
        2. Share real-time whiteboard: User must be able to interact with other users through graphics/drawings in real time.
        3. User Interaction: All users in the same room should be able to draw on the whiteboard in real time.

    - **Non-Functional Requirements**
        1. Privacy: Whiteboard being shared in a room should not be accessible to people in other rooms.
        2. Robustness: In case user disconnects through any issue, user should be able to rejoin the same room.
        3. Performance: Application must be light weight and must not lag.

TECHNOLOGIES USED

- **NodeJS** is a JavaScript runtime built on Chrome's V8 JavaScript engine. As an asynchronous event-driven JavaScript runtime, Node.js is designed to build scalable network applications. In the following
- "hello world" example, many connections can be handled concurrently. Upon each connection, the call-back is fired, but if there is no work to be done, Node.js will sleep.

- **ExpressJS** is Fast, unopinionated, minimalist and flexible web framework or application for Node.js. It provides a robust set of features for web and mobile applications. Creating API’s is also very easy using expressjs. Many popular frameworks like GraphQL are built on Express.

- **Socket IO** is Bidirectional and low-latency communication for every platform. It is Performant - In most cases, the connection will be established with WebSocket, providing a low-overhead communication channel between the server and the client. It is Reliable - In case the WebSocket connection is not possible, it will fall back to HTTP long-polling. And if the connection is lost, the client will automatically try to reconnect. And it is Scalable - as it Scales to multiple servers and send events to all connected clients with ease.

CONCLUSION AND FUTURE SCOPE

There is always a room for improvements in any apps. Right now, this app is just dealing with communication using graphics/drawing.

I would like to conclude that it was amazing experience while making this project. Though some of the things were tough to achieve and implement but finally it has been a great learning. I was able to make the web app look pretty good and decent as possible by keeping in mind various things like Colour Combinations, Fonts, etc. The main objective of making this project easy to use and useful is achieved so far. Reliability was also one of the objectives that I was trying to provide to users. At the end, I can say that my web application is useful for people.

FUTURE SCOPE:

- Can add Account feature like a particular user can create an account and save his/her progress.
- Can add more options to Customise the Pointer properties like opacity, smoothness, etc.
- Other than this, can allow user to Customise the Backgrounds according to their needs.
- Can add Chat feature like users can parallelly chat with other users while they are sketching/ whiteboarding.

REFERENCES

- HTML Canvas API Documentation by Mozilla MDN: https://developer.mozilla.org/enUS/docs/Web/API/Canvas_API

- Node JS Documentation: https://nodejs.org/en/docs/

- ExpressJS Documentation: https://expressjs.com/
- Socket IO Documentation: https://socket.io/docs/v4/