COMP2043.GRP Final Group Report
Shared Whiteboard

Group Members:
BAI Yufei scyyb1
DANG Wenfei scywd1
HE Naixuan scynh1
LUO Chengtao saycl2
XIA Qing scyqx1
YIN Tianzhu zy18734

Supervisor:
Amin Farjudian

Group 13
School of Computer Science
University of Nottingham Ningbo, China

Apr.21 2020

Abstract

With the impact of COVID-19 virus, it is obvious that the online conference system plays an important role in this global catastrophe. This program has been aimed at developing a shared whiteboard system supporting both LaTeX formula and online interaction, including not only “drawing” sharing but also video and audio communication. The shared whiteboard is an online interactive space available for texts, drawings and graphs where each user’s modification is visible to all users in real-time. Although there are massive relative applications on the market, few of them support professional type-in formula in LaTeX notation. On the other hand, this function is necessary for specific users, including experienced experts in Engineering, mathematicians and professors in universities, etc. Therefore, this web-based shared whiteboard application has been developed to meet all the requirements mentioned above.

This report demonstrates the process of developing this application, including system and UI design, implementation, and test and evaluation. Further improvement and reflections are also taken into consideration in the report.

Contents

Abstract **i**

1 Introduction **1**

2 Team Management **3**

3 Technology Background **4**

3.1 HTML5 Canvas 4

3.2 P2P based on WebRTC 5

3.3 Node.js 7

3.4 MathJax 8

3.5 Decisions of Programming Languages 8

4 Project Aim **10**

5 Process **12**

5.1 Development process with ASD 12

5.2 Choice on whiteboard 13

5.3 Three attempts on LaTeX 13

5.3.1 Two failure tries 14

5.3.2 One successful try 16

6 Achievements **17**

6.1 Implementation 17

6.2 User Interface 20

6.3 Interaction of components 22

6.4 List of program achievements 32

7 System tests **36**

8 Reflective Comment **39**

8.1 Technical perspective 39

8.2 Project management perspective 40

9 Summary 42

Bibliography 44

Appendices 45

A User Requirement 45

B System Requirement and Non-functional Requirement 47

C Testing Case 53

D User Manual 62

E Meeting Minutes 67

Chapter 1

Introduction

Shared whiteboard is a whiteboard, but the user can share it online with different people in distinct parts of the world. It refers a to a technology in which a user at a first digital system can write/draw desired content and the same content is simultaneously displayed at a second digital system (Bolia, 2014). A source board refers to the first digital system at which the content originates and a target board, of the (shared) digital whiteboard a target board, of the (shared) digital whiteboard. It enables real-time synchronization of one person's modifications to other devices in a network-based environment, which allows multi-users to communicate with each other by contributing with notes, images, or drawing on their devices and all the members can see it immediately. There is plenty of this kind of software already on the market, including ZOOM, Twiddla and herokuapp, some interactive and collaborative technologies are added to them, including audio/video communication and chat boxes. However, few of them provide expertise communication. None of any existing shared whiteboard supports functionality that allows users to type in a formula in

LaTeX notation, which is an encyclopaedic set of mark-up commands used with TEX, a powerful typesetting program, for the preparation of various types of documents, from scientific articles, reports, to complex books (Kopka and Daly, 2004). Typing or drawing a complicated formula only by hand or keyboard is very difficult. Therefore, our purpose is developing a shared whiteboard which not only provides video communication and text chat but also can render LaTeXstatements into typeset formulas. Although there already exist very well-developed products in the market, our project provides more functionalities that they do not have, which is our project's advantages. For instance, LaTeXfunctionality offers users to present exact mathematical formulas. Since there are multiple open source libraries, our team does not need to develop everything from scratch but can combine and modify those proper libraries with a suitable license. Therefore, our main problem might be how to find the appropriate libraries and how to integrate those components into a whole software.

This report is a final report of our project. Briefly, the report has eight parts, in the first four parts the technology background, team management and project aim will be introduced. After that and overview of the project process and a description of achievement, including the program implementation and user interface. Then, an overview of system testing. Finally, a reflective comment of the project will be mentioned.

Chapter 2

Team Management

Our team is led by Chengtao Luo. We have six team members in total, who are divided into two groups according to distinct assignment:

1. Technology group: Chengtao Luo and Qing XIA

   This group mainly contribute the programming part.

2. Wenfei DANG, Chengtao Luo, Qing XIA, Naixuan HE, Yufei BAI and Tianzhu YIN

   This group mainly contribute writing part.

3. 3. Testing group: Chengtao LUO, Naixuan HE

   This group mainly manage system testing.

Chapter 3

Technology Background

3.1 HTML5 Canvas

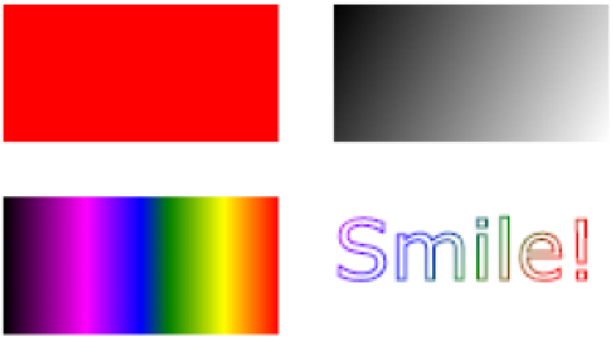


Figure 3.1: *HTML5 Canvas*

The "drawing" function is based on the html5 canvas. This is a html5 element that allows users to draw images controlled by JavaScript. Input drawings, text and pictures in the whiteboard are achieved by it. After creating drawable area in html code, JavaScript code is able to access this area and create new images by integrated drawing functions. One of the benefits the canvas offered is that it is compatible most common browsers, including Firefox 1.5, Internet Explorer (IE9), Chrome and Opera 9+.

By default, canvas element is covered by grid. Every cell in a grid is equivalent to a pixel in a canvas element. The grid starts at the upper left corner (coordinates are (0,0)) and all other elements are allocated from the original point.

Although canvas only offers rectangle graph drawing, other graphs can be created by paths. A path is a collection of points of different shapes formed by connecting lines or curves of different colours and widths. Every path is closed, even for a sub-path. To create a path extra steps are required. First, start point is created. Then drawing methods are called to draw the path and closure is applied right after. Finally, the shape is rendered by filling the path area.

Besides graphs, canvas also provides ways to insert texts and images with shaping and colouring methods. All these functions are feasible and convenient tools to show drawings, text and images (including LaTeX) on the whiteboard. As for the synchronization of the web page, it is established by the WebRTC technique.

3.2 P2P based on WebRTC

In order to achieve the audio/video communication functions embedded in the whiteboard, our team has learnt a lot about the network communication, and finally we decided to use the WebRTC API to implement the functionality. WebRTC (Web Real-Time Communication) is a very simple but powerful API to conduct real-time video/audio conversations on web browsers or mobile terminals. Without downloading or installing any plug-ins, the web developers do not need to pay attention to the digital signal processing process of multimedia, we can just write JavaScript programs to achieve the function. Now the problem is that which network communication mode should we choose, C/S or P2P? In a C/S-based network, clients can be located (requires IP and ports) by registering with the server, and the server holds a one-to-one relation with all users. In a P2P network, direct communication between any two endpoints is possible, each computer has the same function, a computer can be used as a server, set the Shared resources for the use of other computers in a network, and can be used as a workstation, the entire network generally does not rely on dedicated server, and there are no dedicated workstations. Since our group decided to use WebRTC as the communication protocol, therefore, for WebRTC, in CS construction, there are two modes: Mixer and Router. In P2P construction, Mesh is always used. Generally, the Mesh is the simplest mode. It is quite popular among the new WebRTC-provide company, because it does not need to build a server. This mode is based on creating many one-to-one data streams from each sender to each receiver(Figure.3.2).

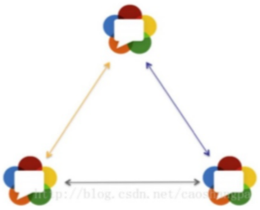


Figure 3.2: *Mesh Mode*

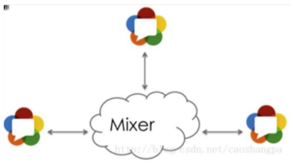


Figure 3.3: *Mixer*

Mixer (Figure.3.3) is the traditional choice for multiple video communication, which has been used for ages and achieves great success. The success is owned to the least cost of the endpoint needs. This construction is based on the server contact with each participant with a single one-to-one stream. When the central unit receives and mixes each exported audio and video stream, it creates an individual stream for each participant. Mixer is an acceptable solution for interoperability with older equipment, which also allows full bit rate adaptation, since mixers can produce different output streams, providing different qualities to each receiver. Another advantage of the Mixer solution is that it can utilize hardware decoding in the equipment as long as the equipment has hardware decoding capabilities.

Router (or Relay) becomes widespread because of H.264 SVC infrastructure, and this construction is used by more and more new WebRTC platform (Figure.3.4). The construction receives flows from each sender based on a central point and sends flows to each participant. This central point can only detect and forward packets without expensive encoding and decoding of streaming media. The common term is SFU. Router offers a cheap and scalable multi-party solution with lower latency than traditional Mixer solutions with video quality. On the other hand, the scheme has less industry experience, making it tricky to adapt the stream to different recipients. Developers need to support the generation of many different streams in the terminal that can then be selectively forwarded in the router. If only audio service is needed, the software needs to interact with old devices and the cost problem does not exist, Mixer would be the best solution. However, this does not fit our actual development situation. Generally, if the software is needed by multiple users, Router should be considered first because Router is more flexible for users. Mesh construction is needed when the number of users is limited. Therefore, our team choose to use Mesh to be audio and video mode.

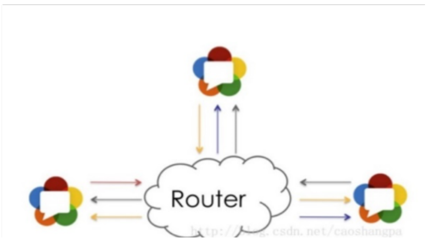


Figure 3.4: *Router*

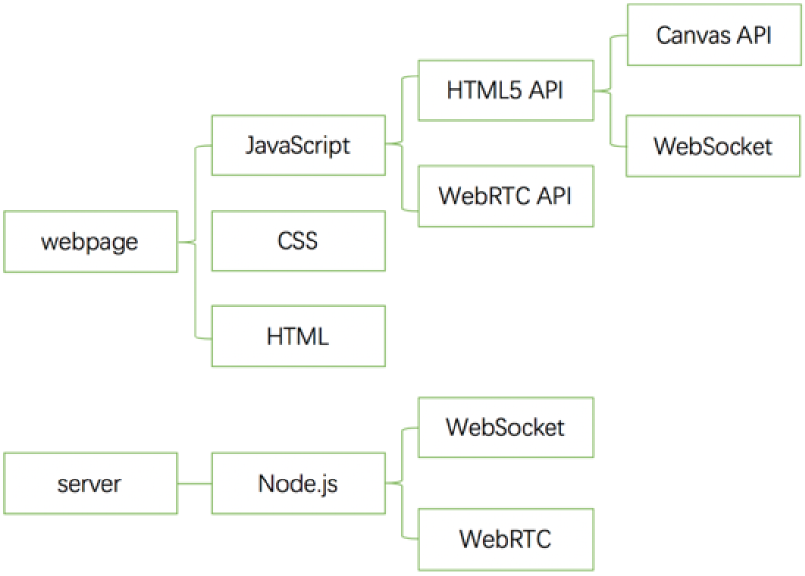
3.3 Node.js

Node.js is an open source platform that allow you to build fast and scal- able network services and applications. It is a JavaScript runtime built on Chrome's V8 JavaScript engine Tilkov and Vinoski(2010). There- fore, it supports the use of JavaScript as a server-side language. In this project, due to the lack of development experience of the team mem- bers, it is difficult to separate the front-end and back-end development. Nodejs enables the front-end and back-end to be used together.

3.4 MathJax

In this sheared whiteboard system, rendering formulae in LaTeXnotation is expected. MathJax is a JavaScript display engine for mathematics that works in all browsers. MathJax enables the front end to support LaTeXmathematical formula and render beautiful styles. It is used in our project for rendering formulae typed in LaTeXnotation. MathJax can use TEX as input and produce SVG as output, in this way users could zoom equations scale randomly at any level. Besides, the entered text is recognized as HTML. MathJax has a rich application programming interface (API), which can be used to make mathematics on web pages interactive. Therefore, in our case MathJax is relatively better choice for our whiteboard is based on HTML canvas. By adding few code lines, LaTeXcan be implemented in our whiteboard.

3.5 Decisions of Programming Languages



Our project is a real-time communication software. Considering that users will use different operating systems or electronic devices, the software is determined to be a web application. Therefore, the user can use the shared whiteboard immediately and free of charge without downloading any software. Because the application is determined to be in the form of a website, our group decided to use HTML and CSS to complete the presentation of the web page. Because the shared whiteboard has many different functional components, we decided to use open source libraries to improve our application. Considering all the programming languages used by library and website format, JavaScript is regarded as the final choice. JavaScript is used to control the function of the webpage, including drawing operation, LaTeXrender, audio and video connection, chat operation and synchronous transmission of information. HTML5canvas API is used for drawing operations and WebSocket is used for synchronous real-time data transmission.

Node.js is adopted as the web server in the background of the system. It sup-ports JavaScript as a server-side language. The main function of WebSocket is to accept and deliver messages. WebRTC is mainly responsible for the real-time transmission of audio and video.

Chapter 4

Project Aim

The idealized shared whiteboard our team manages to realize is users can link to the whiteboard with a url, like what individuals do when they want to access google, they just type the google url. Since it is a web software, the whiteboard can be used in multiple platform if the user can access a browser. When they access the home page of the whiteboard, they can choose to create a new room or join an existed one and create his own user id for the room. The room has a unique id that is created by the user and the room creator can set the room private with adding a password. When users enter the room, they can see tools located at the left, chat box located at the right, whiteboard located at the middle, video image located at the bottom, and they also can see the room id and room url, which users can enter this room through and a symbol showing whether this room is private or not. For the tools, the project provides "pen" with different width, "shape" with different shape, "eraser", "colour" with different colours for pen and shape, "LaTeX" for users to create LaTeXformula by writing LaTeXcode, "screen sharing" and "export" for user to export the content as pdf. For the chat box, users can type words with no length limitation, send emoji, choose a specific user to chat or to public, share files including pictures, books no more than 10m. For video image, users can choose to mute or unmute himself and turn on or off the camera to show himself like zoom. The room creator can banish a specific user and mute everybody.

When the room creator leaves or every user leave, the room will be dismissed, however, the content of the whiteboard will be saved in 10 days.

Chapter 5

Process

5.1 Development process with ASD

Adaptive software development (ASD) is a software development process that focuses on mission, bases on feature and is iterative, limited by time and driven by risk. Besides, ASD provides for continuous learning and adaptation to the emergent state to the project. For the program is a short-term development, it requires every team member learn relevant knowledge and because of COVID-19, the team have to adapt any emergent state. Therefore, our development is based on ASD. In order to realize the idealized project, our team managed to spend twenty-five weeks in total.

At the first week, we implement a market research to have a visualization of what a shared whiteboard is. Zoom and twiddla leave a deep impression on us. At the next two weeks, we discussed details of the project, including the purpose, UI draft and requirements of the program. After that, we used two weeks to take relevant research and find available libraries.

Because of ASD, we could continuous learning during the development. As a consequence, after equipped with enough knowledge, we started to design program features and system structure, and this costed another two weeks. In the next ten weeks, we mainly programmed the system and tried to implement some functions, during this we tested three different whiteboard components, only one of them can be used successfully to our program (detail discussions are in 5.1 to 5.3).

Later, we spent about three weeks to test the system and contributed another four weeks for the system refinements and the final report.

5.2 Choice on whiteboard

Our team, at the beginning of the project, finds three open source shared whiteboard, Matisse is abandoned because the author did not maintain the project for a long time, which leads to that the APIs and libraries that it used are incompatible. Herokuapp is abandoned because the content in the whiteboard will be vanished if everybody leaves the room and cannot be saved and if the creator of the room leaves, the room will be dismissed and cannot be found. The third one, whitebophir is more suitable one compared to the first two whiteboard. It can save the content of the whiteboard according to room id, which means that if every user leaves the room and enter the same room id in the future, they can still enter the same room that is used by them and the content is saved as if they do not leave. Therefore, our team chooses this one to start adding LaTeX functionality.

5.3 Three attempts on LaTeX

For implementing LaTeX feature, we had three attempts in total. Based on ASD, we tested every method for LaTeX to figure out whether it can be used or satisfied user requirement and iterated development. Ultimately, first failed on using specific tag to add elements to svg whiteboard dynamically, the second could be displayed on the whiteboard but did not fit user requirement. Then, after another attempt and test, the last one succeeded (the details are form subsection 5.4.1 to 5.4.2).

5.3.1 Two failure tries

The first one is based on a svg whiteboard. The foundation we used to use was a whiteboard based on svg and use MathJax library to render LaTeX formula. The most common way to render LaTeX in SVG is to use the SVG <foreignObject> tag, which allows non-SVG type element to be embedded in SVG document fragments. It can detect the special delimiter \(\) or \ [\] or $ $, and interprets it as tex mathematical mode. However, ¡foreignObject¿ tag does not support users to add to SVG whiteboard dynamically, it can only render elements that exist in the svg whiteboard, which leads to its inability to support users to edit the contents of formulas.

We hope to have a function that can render formulas in <text>. Finally, we found a method that simulates the conversion of LaTeX code in the ¡text¿ tag into formula rendered by MathJax. This function can copy the text content of the <text> element to a temporary <div> element elsewhere on-screen, so that MathJax can typeset the equation properly. And then it can clone the resulting elements and replace the original <text> element in the SVG with the results of MathJax typesetting. We tried to use it in the whiteboard to generate formulas. Our idea is that the user input can be directly converted into a formula in this way, so that the user can directly add a formula anywhere on the whiteboard. For message transmission, the whiteboard only needs to transmit <text> content, while the formula will be rendered on the user's local whiteboard and will not occupy transmission resources. Every time the user receives a new element, the page is automatically refreshed to render the formula.

When we tested, we found that this method can only be used in local whiteboards. However, the shared whiteboard that we use as the foundation will check the element on the whiteboard during transmission, and the element types that are not generated by the whiteboard will not be displayed. When we tested on the shared whiteboard, unknown errors occurred. The formula shows a translucent state(Figure.5.2), which we think may be related to whiteboard message transmission. We tried to solve this problem, but failed. We could not find out the exact cause of the problem, so we gave up this method of generating the formula and try another way.

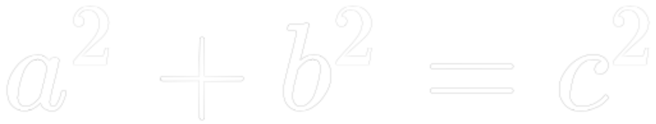
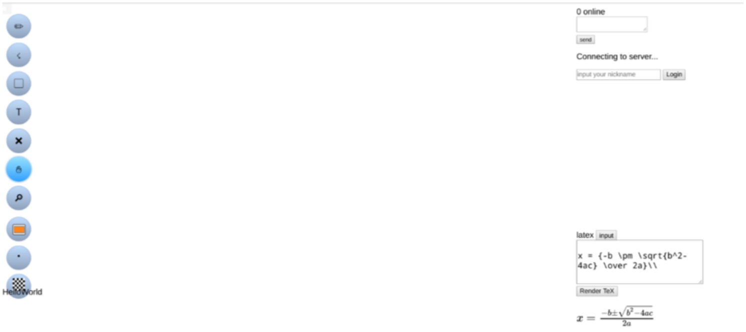


Figure 5.1: *translucent state formula*

Using <text> tag to realize LaTeX functionality will cause an unsolvable problem, which is a translucent state of the formula, our team manages to work on another method, using a box beside the whiteboard to display LaTeX formula. This is done by using MathJax. Using a box to get what users input and store them as picture LaTeX codes in the program, extract the codes and transfer them into string and transfer the string into a dom object of jquery, and use MathJax to transfer them into a text and display the formula in the output box beside the whiteboard. This method is inconvenient for users because the formula is rendered as html element and can just located in the output box, which means that when users want to put the formula next to a picture of a function, they are unable to realize it. Therefore, this method is abandoned because of its inconvenience. For the whitebophir whiteboard, our team is unable to fix the transmission problem. Therefore, svg-base whiteboard is not an option for our team, we have to abandon this whiteboard and use a canvas-base whiteboard.



```
x = {-b \pm \sqrt{b^2-4ac} \over 2a}\\
```

$x = \frac{-b\pm\sqrt{b^2-4ac}}{2a}$

Figure 5.2: *second failure attempt*

5.3.2 One successful try

Since the svg-base whiteboard cannot realize LATEX functionality, our team decides to use an open source shared whiteboard based on canvas, herokuapp. As a result, we succeed in achieving LATEX functionality in this whiteboard and detailed information is in section 6.1.

Chapter 6

Achievements

This part will discussion achievements in details, including implementation, UI design and a list of program achievement. The list is a list of this program achievement according to the requirements of the shared whiteboard system. In the users use whiteboard part, system allows user to use whiteboard on a web page and set all users as guests automatically. In create rooms part, system allows the users to create a unique roomID while the users choose a nickname. In join in rooms part, system allows a user to enter the roomID to join the room. When the users join in a room, system allows users to communicate by using chatting box, video and audio call. In whiteboard use part, system allows user to draw, eraser and type formula on the whiteboard. The details are listed in the list (details in section 6.3).

6.1 Implementation

The shared whiteboard herokuapp is an open source project our team finds in github. It contains complete functionalities except LaTeX.Therefore, our whiteboard is a combination of LaTeX functionality and this open source whiteboard. Based on the foundation of herokuapp, our team manages to utilize the adding picture functionality which the whiteboard provides. Our basic idea is as followed.

1. Provide a typing box that allows user to input LaTeX code
2. Render the LaTeX code and transfer them into LaTeX formula
3. Store the formula as a picture in user's local laptop
4. Use the adding picture functionality to add LaTeX formula.

For step 1, an input box is created which is located above video part and allows users to write LaTeX code with $ as the beginning and the end. The reason will be explained in step 2.

For step 2, after finishing writing codes, a button ”render LaTeXn̈eeds to be clicked, and the rendered LaTeX formula is provided below the button as a picture version. This is done by using MathJax library to render the LaTeX codes and codes need to start with and end with the mark $ in order that they can be detected as LaTeX code and render to LaTeX formula, otherwise they will just become English letters. The input string will be stored in a div tag and a function getElementById in index.js is used to detect whether the button is pressed.

For step 3, storing picture in the local laptop is basic functionality provided by browsers. This is achieved by users.

For step 4, using the functionality adding picture that whiteboard has, add the picture into the whiteboard. The picture can be zoomed or shrink and can be moved to anywhere in the whiteboard.

Using this method to realize LaTeX functionality is not that convenient for users and this web app cannot reach a professional level, however, this is as far as our team goes.

The hierarchy of LaTeX functionality is displaced as a diagram and the explanation of the diagram is as followed. The LaTeX functionality contains three main parts, including "Get user's input", "Transfer to image" and "Add to whiteboard". An input box is provided for users to input LaTeX code and a function getElementById("txtInput") collects the string that users input and a function getElementById("mathjax_result") stores the string in a div tag. For part 2 "Transfer to image" has a function getElementById("btnSave") to detect whether the button "render LaTeX is pressed and if this has been done, the propram calls MathJax, a LaTeX api, to transfer LaTeX code to LaTeX formula as image. After that, a function "fileWriter" is called to displace the image below the input box. Part 3 is achieved by users. "Save the image in local space" is a functionality provided by browser and users can right click the mouse to achieve it, and "Import the image" means using the functionality that the whiteboard provides to import the LaTeX image to the whiteboard.

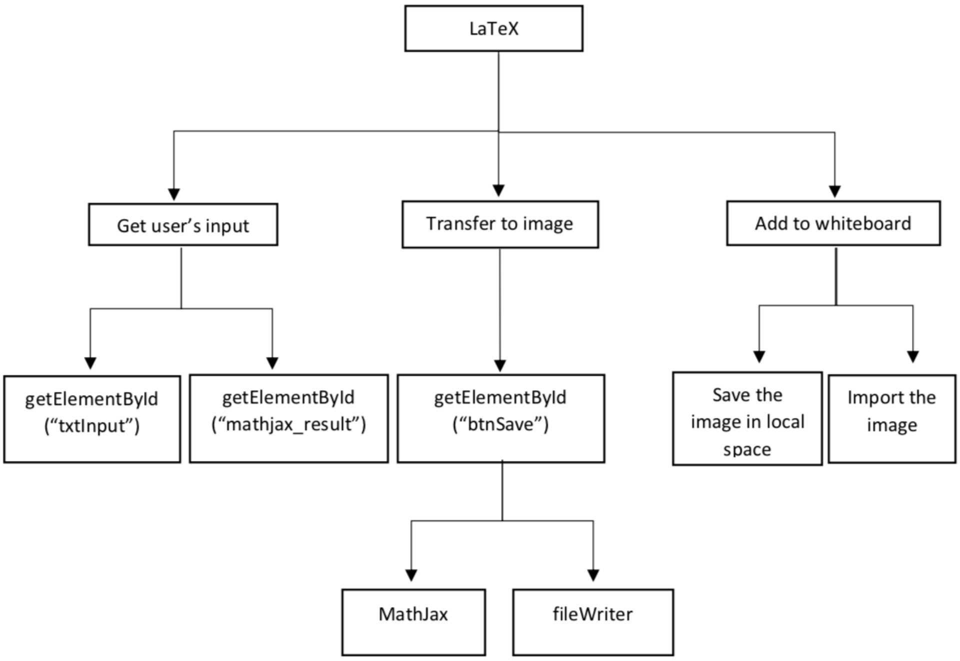
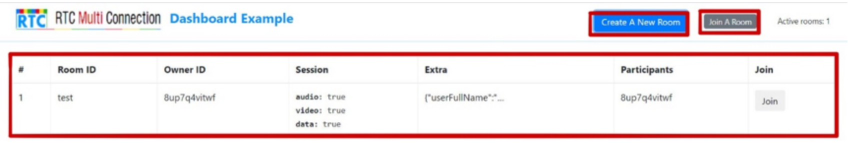

LaTeX
Get user's input
Transfer to image
Add to whiteboard
getElementById ("txtInput")
getElementById ("mathjax_result")
getElementById ("btnSave")
Save the image in local space
Import the image
MathJax
fileWriter

Figure 6.1: *LaTeX function structure*

6.2 User Interface

1. Home Page



#	Room ID	Owner ID	Session	Extra	Participants	Join
1	test	8up7q4vitwf	audio: true video: true data: true	{"userFullName":"...	8up7q4vitwf	Join

Figure 6.2: *The home page for creating or joining room*

On the right top corner, there are two buttons, users could choose to create a room or join a room by clicking the corresponding button. Meanwhile, users could check the number of active rooms on this corner.
In the middle of the screen there will be a table of room, which has been created and the relevant information of it, including room owner ID, session and participants ID. Users could view the table and choose the specific one to join in by clicking the “Join” button in the table.

2. Whiteboard Page

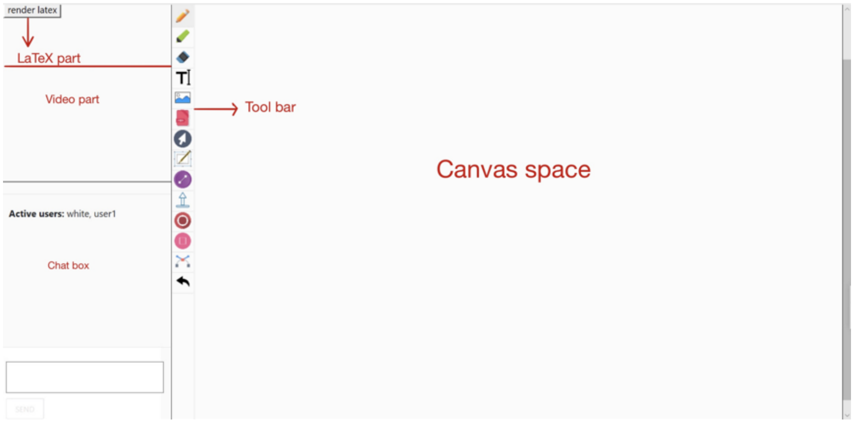


Figure 6.3: *Inside room whiteboard page*

Tool Bar: The toolbar on the left of the canvas region allows the user to choose different editing tools. The currently used tool will be shaded.
Shared canvas spaces: The canvas space displays the contents of the current page. The behavior of mouse actions in the drawing space is dependent on the current active mode.
LaTeX part: To render latex codes into formula.

Video part: To display all members' video images.

Text chat box: Allows users to send text and files, and the chat history is displayed above.

6.3 Interaction of components

The whiteboard system is divided into two parts: room management and whiteboard. Room management involves creating and joining rooms. And the whiteboard mainly contains four components including the LaTeX part, video/audio chat, text chat box and a functional canvas space.

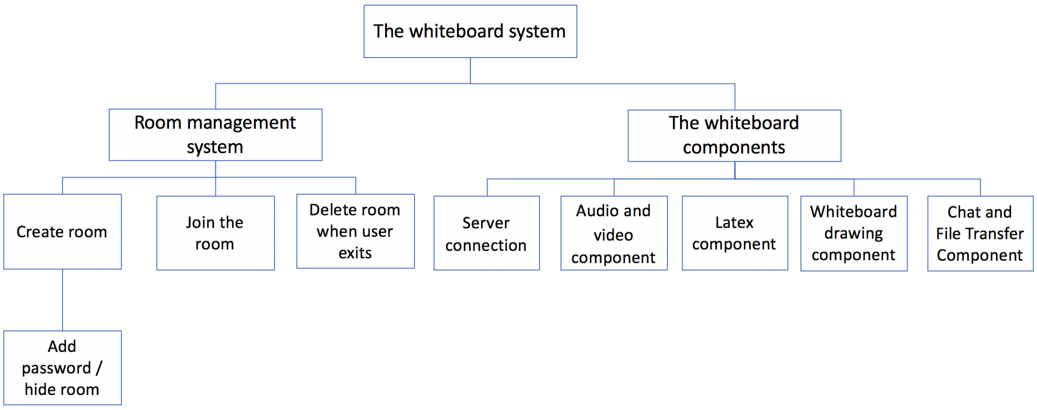

The whiteboard system
Room management system
The whiteboard components
Create room
Join the room
Delete room when user exits
Server connection
Audio and video component
Latex component
Whiteboard drawing component
Chat and File Transfer Component
Add password / hide room

Figure 6.4: *Whiteboard component structure*

A great advantage of the shared whiteboard is its interactivity, the diagram below will show the components interaction.

Room management

1. Create a new room

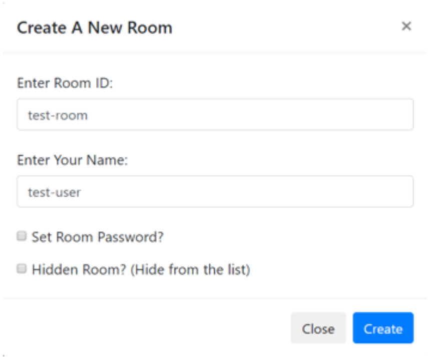
   Room ID: Set the ID of the room.

   Enter Your Name: Set the nickname for meeting.

   Set Room Password: Set the password for the meeting room(optional).

   Hidden room: Hide the room form the homepage table(optional).

   Create/close: Create the room or quit the creation step.



Create A New Room

Enter Room ID: test-room

Enter Your Name: test-user

□ Set Room Password?

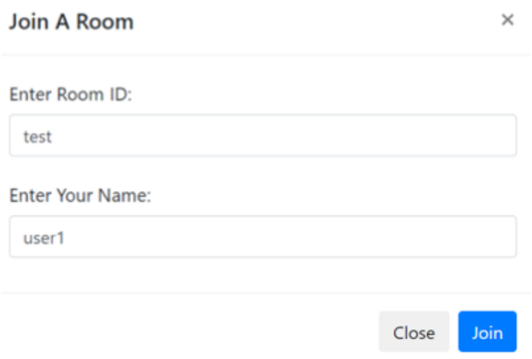
□ Hidden Room? (Hide from the list)

Figure 6.5: *Room creating window*

2. Join A Room: Enter Room ID: Enter the room ID users are looking for

   Enter Your Name: Set the name for meeting

   Join/Close: Join the room if it is found or close the Join window.



Join A Room

Enter Room ID: test

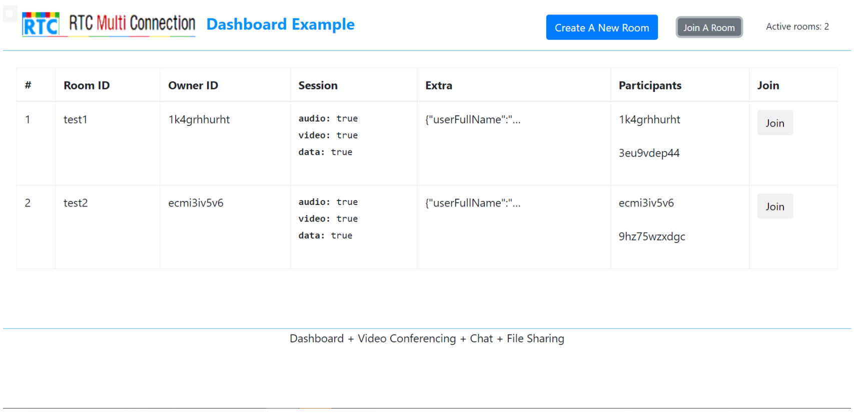
Enter Your Name: user1

Figure 6.6: *Room creating window*

3. Room Table:

   Show the information of all visible rooms.

   Join: Set the username and join this room.



#	Room ID	Owner ID	Session	Extra	Participants	Join
1	test1	1k4grhhurht	audio: true video: true data: true	{"userFullName":"...	1k4grhhurht 3eu9vdep44	Join
2	test2	ecmi3iv5v6	audio: true video: true data: true	{"userFullName":"...	ecmi3iv5v6 9hz75wzxdgc	Join

Dashboard + Video Conferencing + Chat + File Sharing

Figure 6.7: *Room creating window*

Whiteboard components

1. Canvas painting and marking.

   These interactions are mainly achieved by these three tools.

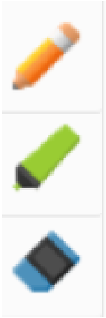


Figure 6.8: *Painting and marking tools*

1.1 Pencil: a blue pencil to draw or write things.

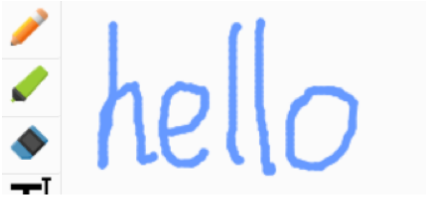


Figure 6.9: *Pencil tool demo*

1.2 Marker: a pink marker to mark things.

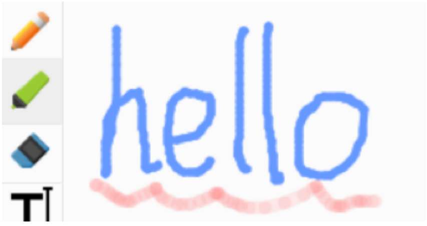


Figure 6.10: *Marker demo*

1.3 Eraser: act as a white colour pen to overlay other images, but the colour is slightly different from the background.

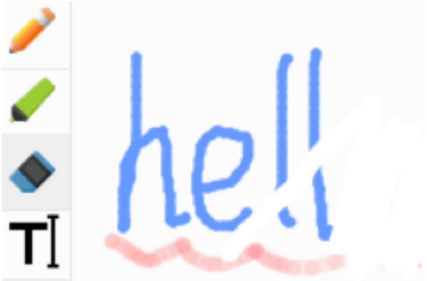


Figure 6.11: *Eraser demo*

2. Text and image insertion.

   Mainly achieved by these tools.

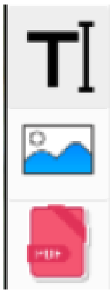


Figure 6.12: *Texting and adding images tools*

2.1 Write text: click to set the font and type the text anywhere in canvas.

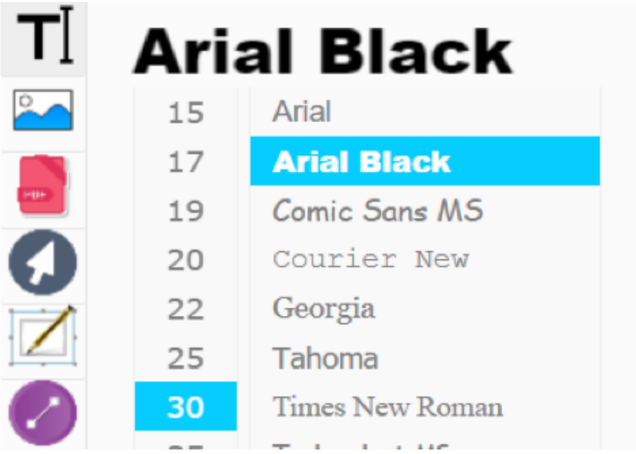


Figure 6.13: *Text adding demo*

2.2/2.3 Add image and Add pdf: These two work similarly, click the button and choose the appropriate file been added. The size of images can be justified.

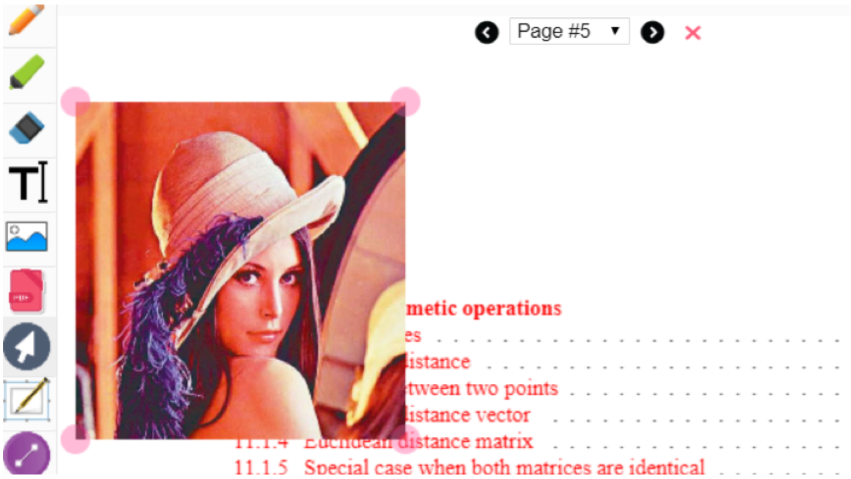

Page #5
metic operations
es
istance
tween two points
istance vector
11.1.4 Euclidean distance matrix
11.1.5 Special case when both matrices are identical

Figure 6.14: *Image and PDF file demo*

3. Geometric shape.

   The whiteboard can generate several common geometric figures by these tools.

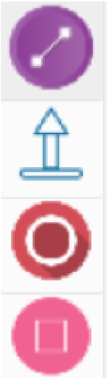


Figure 6.15: *Tools bar*

3.1 Draw lines: click twice to set a lines two endpoints.

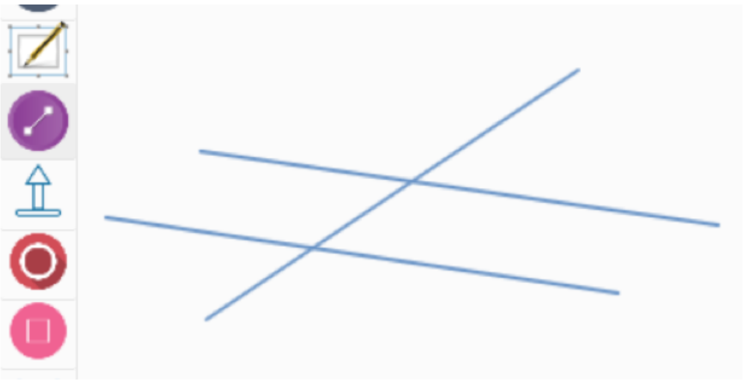


Figure 6.16: *lines drawing demo*

3.2 Draw arrows: click twice to set the end points.

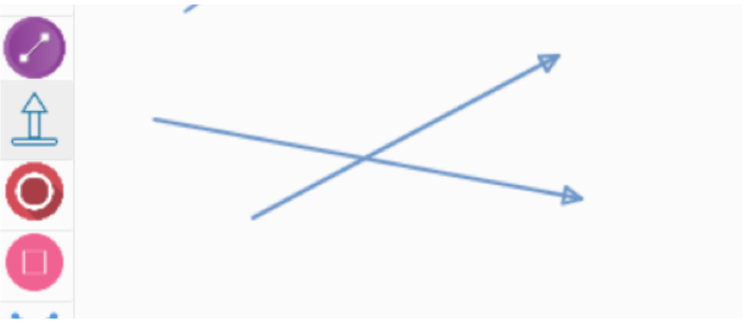


Figure 6.17: *arrows drawing demo*

3.3 Draw arc/rectangle: click once and release to set the size.

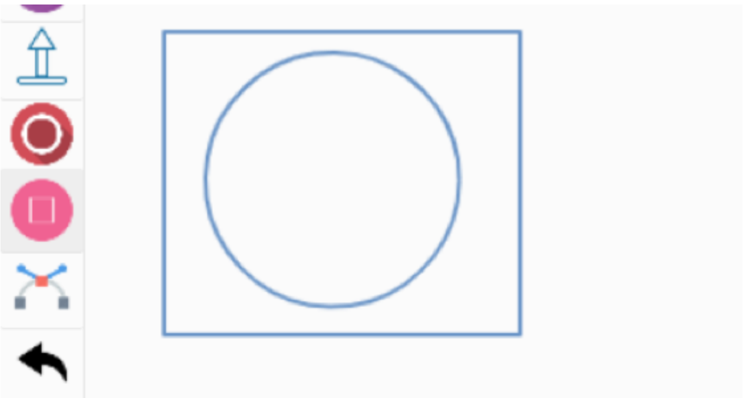


Figure 6.18: *arc/rectangle demo*

4. Adjust the content.

4.1 By these tow button, the last stroke or all contents can be moved in the canvas.

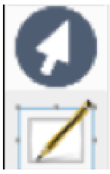


Figure 6.19: *related tools*

4.2 The undo button can remove the last stroke.

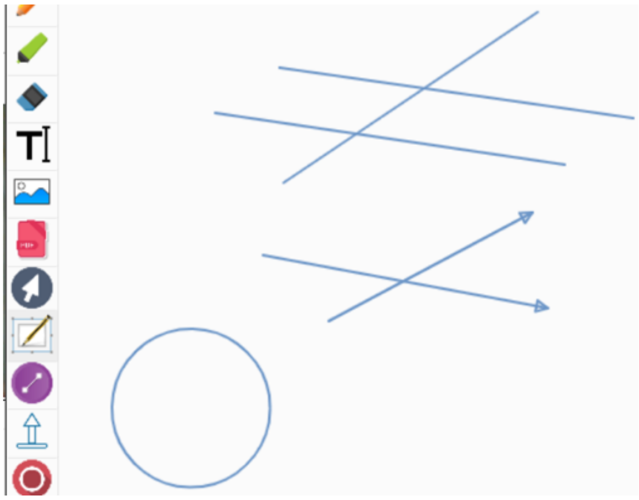


Figure 6.20: *moved from the Figure.6.17 and Figure.6.18*

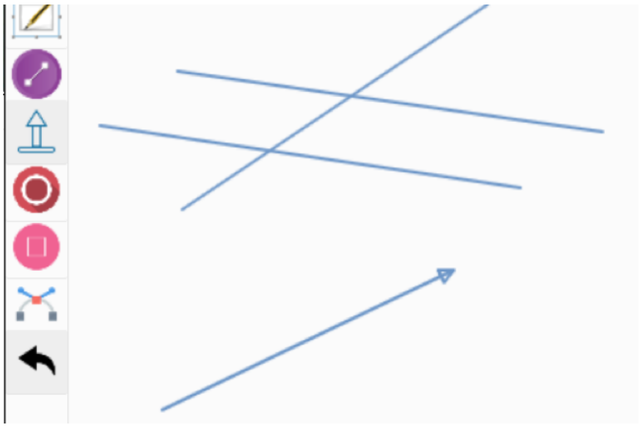


Figure 6.21: *An arrow removed from Figure.6.20*

5. The video panel. You can pause the video; adjust one's voice or the position of the video window (full screen or floating window)

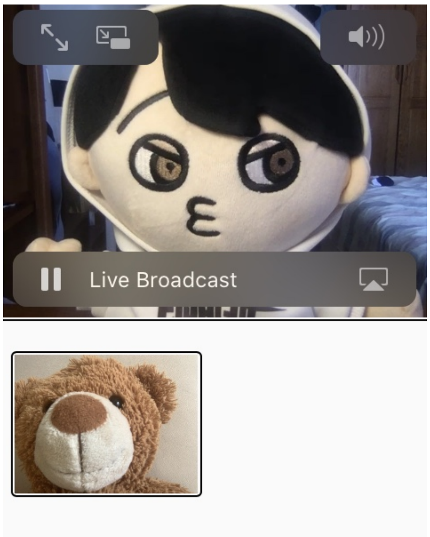
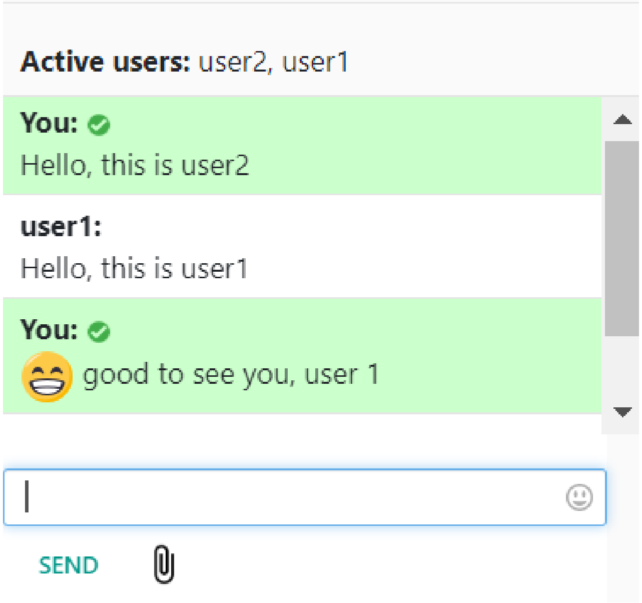


Figure 6.22: *Video panel*

6. Chatting box

6.1 Type in the message in the dialog and click SEND button to send it.



Active users: user2, user1

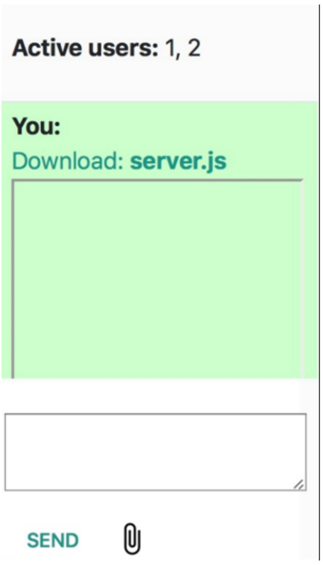
You: ✓
Hello, this is user2

user1:
Hello, this is user1

You: ✓
😁 good to see you, user 1

Figure 6.23: *chatting box demo*

6.2 Files can be shared by clicking the clip button, other members in the meeting room can download them.



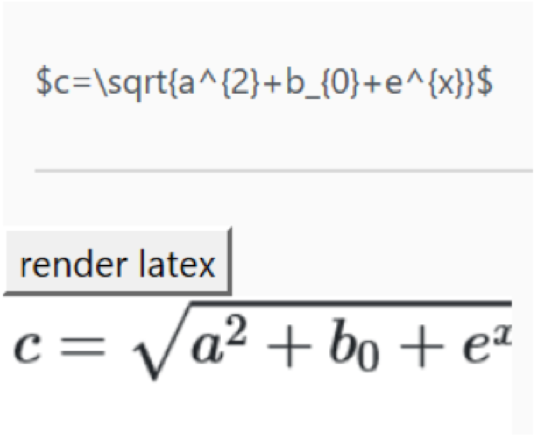
Active users: 1, 2

You:
Download: **server.js**

Figure 6.24: *File sharing demo*

7. LaTeX part: To render latex codes into formula.

   Render LaTeX: to render the input LaTeX into formula image.



```
$c=\sqrt{a^{2}+b_{0}+e^{x}}$
```

$$c = \sqrt{a^2 + b_0 + e^x}$$

Figure 6.25: *A LaTeX formula rendering demo*

6.4 List of program achievements

In the latest version of the whiteboard, all essential requirements have been reached. Some additional requirements have been reached as well. The following list shows the achievement for now.

1. Use the whiteboard:

   The whiteboard can be used on a website.

2. Create a room:

   ✓There is no login process, which means all users are set as guests automatically.

   ✓A room ID can be created by the user and a password (optional) as well.

   ✓A nickname is asked to be set when the user enters the room.

   ✓The system can detect whether input IDs /passwords have been used and will send a warning message if it happens.

   ✗The system cannot detect the same nickname of different users.

3. Invite others:

   ✓System allows a user to enter the room ID to join the room.

   ✗System allows a user to invite other users to join the room by sharing the room URL.

4. Join a room:

   ✓System allows a user to enter a room by choosing one in the room list.

   ✓System allows a user to enter the room ID to join the room

   ✓Room creator can hide the room so that others cannot see the room in the list.

   ✗System allows users to re-enter the room with latest whiteboard image.

5. Use chat box:

   ✓Sending text message (no more than 120 characters). If exceeds, a warning would display.

✓System allows a user to review the previous message.

✓System allows a user to identify a message sent by others by nickname showing with that message.

✓Users can upload any types of files.

✓System allows users to use Emoji.

✗System allows a user to see the time stamp.

6. Use video/audio call:

   ✓System sends permission message to ask for the camera or microphone permission.

   ✓System allows users to communicate by using video/audio call.

7. Draw and erase on Whiteboard:

   ✓System allows a user to draw on the Whiteboard by using a mouse.

   ✓System allows a user to draw on the Whiteboard by using a digital pen.

   ✓System allows a user to choose different types of pens (2 types).

   ✓System allows a user to choose different shapes to insert on the Whiteboard, including rectangle and circle.

   ✓System allows a user to use eraser to eraser what he/she drew on the Whiteboard.

   ✓System allows a user to undo his/her instructions.

   ✗User can redo their instructions.

   ✗Users can choose different colours of the pen (each member only has one unique colour).

   ✗Whiteboard can be zoomed.

   ✗Room creator can proscribe any other room member's actions on the Whiteboard.

   ✗Users can clear the whole whiteboard.

8. LaTeX:

✓System allows a user to type formula in LaTeX notation to the LaTeX box.

✓System allows a user to get the formula in a form of image by clicking "Render LaTeX" button.

✓System allows a user to generate multiple formulas.

✓System allows a user to drag the formula, which is generated by himself/herself, to anywhere on the Whiteboard.

✓System allows a user to reedit the generated formulas as long as this is his/her last move.

✗Users can view formula history and re-edit it.

9. Insertion:

   ✓System allows a user to upload PDF files and drug to any area on the Whiteboard.

   ✗Exportation is available in current version.

10. Reliability and security:

    ✓The system should check the input data to prevent abnormal operation.

    ✓System needs no security vulnerabilities in programs.

    ✗The system needs to ensure the safety of user information.

11. Easy to use:

    ✓System can be used directly without downloading any software.

    ✓System needs to simplify the operation mode of application and make it easy to learn and master.

    ✓The system interface design should be simple and clear.

    ✓The system only needs server-side update.

12. Performance requirements:

    ✓The system needs to ensure the concurrency of information transmission.

✓Response time can be checked in backstage.

✓System needs to ensure smooth and stable when the number of users is large (no more than 8 users).

13. Extensibility and maintainability:

    ✓The system allows expansion for different over-capacity requirements.

    ✓The system needs to conform to the design pattern to facilitate later maintenance.

14. Compatibility:

    ✓The system should support users to use Windows.

Chapter 7

System tests

All tests (details are in the appendix) have been measured and evaluated. In general, most tests were passed successfully.

1. Creating the room:

   Users can open the whiteboard website by currently popular browsers including Chrome, Firefox, IE and Edge. Rooms can be created by setting a room name and password (optional). In addition, room can be hidden from the list so that it can only be joint by searching the name. If users enter wrong input, warning messages would display. Every participant needs to choose a nickname before entering.

2. Whiteboard functions:

   Users can draw by two types of whiteboard pens while the colour of those two pens cannot change. Texts, graphs (circles, rectangles, arcs and arrows) and pictures can insert to the whiteboard and the location and size can be adjusted. Besides, PDF files can also be uploaded to the whiteboard. The eraser can erase any drawings or graphs on the whiteboard. In fact, it is equal to a white pen.

3. Texting function:

   Text messages can be sent by clicking the send button. Emojis can be sent as well. Every message display with the sender's name while the timestamp display is not available currently. Moreover, any type of files can be uploaded to the text box. All the history messages or files can be viewed by dragging the side bar.

4. Software stability:

   For the test of maximum users, due to the CODIV-19 influence, the biggest pressure test only takes 3 users, which has no effect to the smoothness of the software. For the reconnection test, users can reconnect to the room with proper video and audio, and the whiteboard displays the current image, but texting records are lost.

5. Invitation:

   Users can join a room by the room list at the home page or searching the room name by clicking the "join a room" button. If the room creator chooses to hide the room, users can only join the room by searching the name.

6. Video and audio:

   Once users permit the usage of cameras and speakerphones, video and audio operate properly, while users can only adjust the first video window's volume. However, there is a defect that if user prohibited the usage of camera and audio, the room he creates or the one joins is a room that does not exist. Neither others can search this room, nor he/she can successfully join the room. Only himself/herself is in this "ghost" room.

7. Undo:

   The undo function can withdraw any last instructions including drawing, insertion or movement. This undo command will withdraw the last instruction from the last modified user, no matter it is the user of this function or other users. In another word, users may undo other user's instruction.

8. LaTeX:

   Users can input correct format text to generate a picture of LaTeX formula. Then users can download the picture to the local and re-insert it as inserting a picture. The adjustment of the formula is identical to the adjustment of inserted images.

Chapter 8

Reflective Comment

8.1 Technical perspective

During this program developing, we learnt a lot of relevant knowledge, such as HTML5 canvas, WebRTC, node.js, etc. Furthermore, to have access to these pieces of knowledge, we also gained the ability of self-learning and online resource searching. In fact, self-learning is a useful personal skill, since nobody could be able to stay in school and gain knowledge from others' teaching for a life-time. Hence, this project was a great chance to practice it for everyone on the team. Simultaneously, to ensure a collaborative programming process, we also learnt how to use GitHub efficiently. By using this platform, telecommuting could be carried out between team members and any changes of codes could be shared among the team in real-time. With the help of professional tools, teamwork could process smoothly. As a consequence, the effectiveness of the program was increased and this was the key to allowing the program could be finished in time.

However, there exist some barriers to the program. One difficulty is that some part of components is not easy to handle, since we used several open sources, understanding codes from other teams would be hard. Most of these are enormous, try to learn every piece of them would be time-wasting. The other is that, because of COVID-19, all team members are under quarantine, in other words, we cannot go out or head to some place together. Moreover, due to our technical limitation, only when different devices connecting to the same network, they can communicate with each other through the whiteboard. Therefore, it is difficult to test whether some features of whiteboard could work as expected under this circumstance. Nevertheless, we tried several ways to overcome the difficulties, one way is that team members connected school network via VDI, the other is one member used different machines to connect the same local network at home to do a performance test. All these methods are to ensure that team members could act as distinct users and utilize the whiteboard in same local area network, in this way system testing can be processed. Eventually, we could accomplish the testing part.

8.2 Project management perspective

Communication is the basement of teamwork. However, we failed on this part at the beginning stage of the program. At that time, the biggest problem of our team was insufficient communication. For instance, online messages could not be replied in time, the discussion on a specific subject could be time-wasting or because of dissimilar personal schedule, it was hard to find a common time an informal meeting. Hence, this problem slowed down the processing of the project for a while. Once we noticed that it should not be ignored anymore and we need to take this seriously, we immediately decided to hold an informal meeting to figure out ways to solve it. Finally, we improved the efficiency of communication among the team by encouraging everyone to get involved in the discussion, increasing numbers of meeting, decreasing the meeting time and trying our best to avoid online communication when team members were all on the campus. However, during vacation and quarantine, we had to discuss via the internet. Then, to overcome this, we communicated by using audio call instead of texting to each other which is inefficient. Furthermore, we learnt that communicating in person is more efficient that online chatting and audio discussion is more efficient than texting conversation.

Meanwhile, there was another problem that the division of labour within the group was not clear, which caused that members were confused about what should do next, and this delayed the process of the project. After realizing the seriousness of this, we changed the way to assign works. Discuss details of tasks together first, then list all assignments and corresponding team member on the paper instead of in an oral way. As a result, the organization of team work is improved, therefore, ensure the project could work as planned.

To summarize, we might encounter many problems or difficulties through this project. However, we always found some ways to solve them. Besides, thanks to this project and unexpected circumstance, we were able to practice the ability of self-learning, overcome some barriers and gain the awareness of the importance of communication and organization from this project.

Chapter 9

Summary

In October last year, our team was assigned to create an online shared whiteboard program with LaTeX. In the past six months, a market survey was made to understand the marketing background of online shared whiteboard at first, and then our team learned about HTML5 canvas, WebRTC, node.js, MathJax and other technologies background through books and network, and finally designed and completed the program that almost meets the requirements. The main functions of this program are: the user can use whiteboard on the web page, the user can create a new room and use a nickname to enter, the user can enter the designated room through a roomID, the user can use chat box to send information and view historical information in the room, the user can use voice and video communication in the room, and the user can use whiteboard with LaTeXin the room. Because of the COVID-19 coronavirus pneumonia epidemic that started in January this year, our team failed to return to school on time to complete the program together, and because of the communication problems caused by time difference and network problem, many of the functions which envisaged in the mid-term work are not realized, and the final results are different from the initial work expectation. In view of the problems existing in our team in this work, our team members think about, analyse and improve them, which will significantly improve our teamwork efficiency in the future.

In the perspective development of the project, LaTeXfunctionality can be realized more automatically if the generated LaTeXimages can be dragged to the whiteboard rather than completing a tedious procedure of storing image in local and import it in whiteboard or LaTeXformula can be written in the whiteboard using a text functionality. Moreover, the project has some functions that can be completed, for instance, pencil functionality can be used more smoothly and users can export whiteboard content. In terms of cooperation work, there is also a lot of space for improvement in the distribution of work contents and the efficiency of communication among team members. In terms of cooperation work, there is also a lot of space for improvement in the distribution of work contents and the efficiency of communication among team members.

Bibliography

Priyank Bolia. Shared digital whiteboard. https://patents.google.com/patent/US20140040767A1/en, 2014.

Helmut Kopka and Patrick W. Daly. *A Guide to LaTeX and Electronic Publishing*. Addison-Wesley, 4th edition, 2004.

Stefan Tilkov and Steve Vinoski. Node. js: Using javascript to build high-performance network programs. *IEEE Internet Computing*, 14(6):80–83, 2010. doi:10.1109/MIC.2010.145.

Appendix A

User Requirement

1. User shall open the Whiteboard on website
2. User shall sign in as a guest
3. User shall set their nickname
4. User shall change their nickname
5. User (room-creator) shall set room password
6. User (room-creator) shall lock the room
7. User (room-creator) shall proscribe other users
8. User shall create a new Whiteboard room
9. User shall join a current room by room number
10. User shall join a current room by room number and password
11. User shall chat with each other

    -Texting
    - File sharing

12. User shall use a video call
13. User shall use an audio call
14. User shall mute all the other users
15. User (member, room-creator) shall write and draw on Whiteboard

    -Use a digital pen
    - Choose multiple colours
    - Insert geometric figure
    - Ease content that has written by himself/herself
    - Insert pictures
    - insert texts
    - insert LaTeX notation formula
    - zoom the Whiteboard
    - undo his/her instructions
    - redo instructions only by himself/herself
    - re-edit the text
    - re-edit the LaTeX notation formula
    - choose and drag a certain area

16. User (room-creator) shall write and draw on Whiteboard - erase anything
17. User shall export the current whiteboard in his/her local computer

    - PDF

  - JPEG

18. All user shall view any changes on the Whiteboard simultaneously
19. User shall use the software for free

Appendix B

System Requirement and Non-functional Requirement

Shared Whiteboard System Requirements		
Requirements		Description
Use Whiteboard	System allows user to use Whiteboard on a website	
	set all users as guests automatically	
Create a room	System generates a unique random roomID	
	System allows a user to create a unique roomID	If the roomID does not exist, system creates a new room. Else, user automatically joins the room already exists.
	System allows a user to choose whether to set a password when creating a room	If the room creator sets a password of it, other users are asked to enter the correct roomID and correct password before joining room. Else, other users join the room only by entering correct roomID
	System allows a user to choose a nickname	If the chosen nickname does not exist in the room, user joins room with that name. Else, system sends a warning message and asks user to re-enter
Invite other users	System allows a user to invite other users to join the room by sharing the room URL and the relative roomID, this user could be the creator of the room or members	

Shared Whiteboard System Requirements		
Requirements		Description
Join a room	System allows a user to enter the roomID to join the room	1.If user enters existed roomID, he/she join the room automatically. Else, system creates a new room with the roomID which does not exist yet
		2. If room has password, user is asked by system to enter the password. Else, user joins the room
		3. If user enters correct password, user joins the room. Else, system sends an error message and asks user to re-enter the password
	System allows a user to choose a nickname	If the chosen nickname does not exist in the room, user joins room with that name. Else, system sends a warning message and asks user to re-enter
Use chat box	System allows users to communicate by using chatting box	
	System allows a user to send text message (no more than 120 characters)	If user type in a message within 120 characters, system sends it to the chatting box. Else, system sends an error message and asks user to re-enter
	System allows a user to upload different types of files (no more than 10MB) by clicking the “Add File” button	If the size of file is less than 10MB, it uploads the file. Else, system sends an error message and asks user to re-choose a file to upload
	System allows a user to review the previous message history	
	System allows users to use Emoji	
	System allows a user to see the time stamp	
	System allows a user to identify a message sent by others by nickname showing with that message	
	System allows users to communicate by using video/audio call	

Use Video/audio call

Shared Whiteboard System Requirements		
Requirements		Description
	System sends permission message to ask for the camera or microphone permission	1.If user local machine has camera and microphone, system allows users give the permission of camera or microphone or both of them. Else, user can not join video/audio call
		2.If user confirms the permission, video or audio starts automatically. Else, only video or audio starts or user refuses to join video or audio call
	System allows users to mute others video/audio channel as well as his or hers	
Draw and erase on Whiteboard	System allows a user to draw on the Whiteboard by using a mouse	
	System allows a user to draw on the Whiteboard by using a digital pen	
	System allows a user to choose different types of pens, including pencil, highlighter and ballpoint pen, etc.	
	System allows a user to choose different colours of the pen, including red, blue and black.	
	System allows a user to choose different shapes to insert on the Whiteboard, including triangle, rectangle and circle.	
	System allows a user to choose what colour to fill the inserted shape.	
	System allows a user to use eraser to eraser what he/she drew on the Whiteboard	

Shared Whiteboard System Requirements		
Requirements		Description
	System allows a user to undo his/her instructions	If user press "un-do" button, system allows user to choose which instruction, only ordered by himself/herself, want to undo. If user choose an instruction ordered by other user, system sends an alert information
	System allows a user to redo his/her instructions	If user press "un-do" button, system allows user to choose which instruction, only ordered by himself/herself, want to undo. If user choose an instruction ordered by other user, system sends an alert information
	System allows a user to zoom Whiteboard	
	System allows a user(room-creator) to proscribe any other room remember having any action on the Whiteboard	
Type formula in Whiteboard	System allows a user to type formula in LaTeX notation	
	System pops up a box to let a user input Latex codes when he/she clicks the "LaTeX" button	
	System allows a user to input formula codes for arbitrary length	
	System allows a user to get the formula in a form of image by clicking "Render TeX" button	
	System allows a user to generate multiple formulas	
	System allows a user to drag the formula, which is generated by himself/herself, to anywhere on the Whiteboard	
	System allows a user to re-edit the generated formulas	
	System allows a user to view formula history and re-edit it	

Shared Whiteboard System Requirements		
Requirements		Description
Clear the Whiteboard	System allows a user(room-creator) to clear the Whiteboard.	System sends a confirming message to confirm whether the user dose want to clear the Whiteboard when one clicks the "Clear" button
Insert	System allows a user to insert multiple types of files into the Whiteboard (no more than 10MB), including pictures and texts	If the size of file is less than 10MB, it inserts when user enter "Insert" button. Else, system sends an error message and asks user to re-insert
	System allows a user to choose which file in local machine to insert	
	System allows a user to choose any inserted file and drug to any area on the Whiteboard	
Export	System allows a user to export current screen of the Whiteboard	
	System allows a user to save the exported screen of the Whiteboard	
	System pops up a window to allow a user to choose to export the current screen as PDF file, JPG file or PNG file and choose the direction to save.	
	If the direction path exists, system saves the file when user clicks "Save" button. Else, system sends an error message and asks user to re-choose the path	
	System automatically generates a name of the exported file or allows user to type in a name	
	If the name does not exist yet, system saves the file when user clicks "Save" button. Else, system will send a warning message and asks user to re-enter a name. Else if, user does not change the name, system saves the file and overwrites the one with the same name	

Non-functional Requirement	
Reliability and Security	System needs no security vulnerabilities in programs
	The system needs to ensure the safety of user information
	The system should check the input data to prevent abnormal operation
Easy to Use	System can be used directly without downloading any software
	System needs to simplify the operation mode of application and make it easy to learn and master
	The system interface design should be simple and clear
	The system only needs server-side update
Performance Requirements	The system needs to ensure the concurrency of information transmission
	Response time
	System needs to ensure smooth and stable when the number of users is large
Extensibility and Maintainability	The system allows expansion for different over-capacity requirements
	The system needs to conform to the design pattern to facilitate later maintenance
Compatibility	The system should support users to use Windows

Appendix C

Testing Case

Test ID	Test Classification	Test Description	Execution Step	Expected Output	Pass / Fail
1	Creating / Joining a room	A room can be created by setting room name and username.	A user creates a room with a room name and a user name.	Room is created with the written room name and user get the username.	Pass
2		A room with password can be created.	Room creator sets a password when creating a room.	The Room is set to have a password.	Pass
3		Users can join a room through url.	User input the url in the browser.	Users join the room.	Fail
4		A room can be joint by its room name.	Users input an existing room name to join a room.	Users join the room.	Pass
5			Users input a new room name to join a room.	Users cannot join the room and an error message is alerted.	Pass

Test ID	Test Classification	Test Description	Execution Step	Expected Output	Pass / Fail
6		A room can be hidden in the list.	Room creator hide the room when creating it	Other users cannot find the room in the index list.	Pass
7		Users can join a hidden room.	Users input the correct hidden room name to join the room.	Users enter the hidden room	Pass
8			Users input the wrong hidden room name to join the room.	"Room is not created" is alerted and Users cannot enter the room.	Pass
9		Users can join a private room.	Users input correct password to join the room.	Users enter the private room.	Pass
10			Users input wrong password to join the room	"Password is wrong" is alerted and Users cannot enter the room.	Pass
11		Warning messages would display when illegal room name is detected	User inputs the same room name to create a new room	"Room has been created" is alerted and user cannot create the room.	Pass
12	Connection	New users can see previous content of whiteboard	New Users enter an existing room.	Users can see previous content of whiteboard.	Pass

Test ID	Test Classification	Test Description	Execution Step	Expected Output	Pass / Fail
13		New users can see previous chat content.	New Users enter an existing room.	Users can see previous content of chatting box.	Fail
14	Reconnection	Users can re-enter the room.	Old users enter the room without re-input username.	Old users enter the room directly.	Fail
15		Room creator re-enter the room.	Room creator enters the room without re-input username.	Room creator enter the room directly.	Fail (The room is dismissed when room creator quiz)
16		Video is still connected when users turn on the dormant laptop	User keeps connecting to the room when he let laptop sleep and open it	Video is connected.	Fail
17		Audio is still connected when users turn on the dormant laptop	User keeps connecting to the room when he let laptop sleep and open it	Audio is connected.	Pass
18		Whiteboard is still connected when users turn on the dormant laptop	User keeps connecting to the room when he let laptop sleep and open it	Whiteboard is connected and previous content can be seen.	Pass

Test ID	Test Classification	Test Description	Execution Step	Expected Output	Pass / Fail
19		Chat is still connected when users turn on the dormant laptop	Room creator keeps connecting to the room when he let laptop sleep and open it	Previous chatting content is seen by room creator.	Pass
20			Other users keep connecting to the room when he let laptop sleep and open it	Previous chatting content is seen by other users.	Fail
21	Painting	Users can use draw lines and curves on the whiteboard.	Users click pencil or mark pen to draw.	Lines and curves are drawn on the whiteboard.	Pass
22		Users can use pencil / mark pen with different width.	User right click pencil / mark pen button to choose different width.	Pencil / mark pen can be written in different width.	Fail
23		Users can use pencil / mark pen with different colors.	User right click pencil / mark pen button to choose different colors.	Pencil / mark pen can be written in different colors.	Fail
24		New lines can cover the old ones.	One user draws lines covering the old lines.	Old lines are covers by the new one.	Pass

Test ID	Test Classification	Test Description	Execution Step	Expected Output	Pass / Fail
25			Different users draw lines covering the old lines.	Old lines are covers by the new one.	Pass
26	Text	Users can text on the whiteboard with different font sizes and font family.	Users click text button and choose font size or family and text.	Different font sizes and different font family text are shown on the whiteboard.	Pass
27	Shapes	Users can insert circles and rectangles on the whiteboard.	Users click circle or rectangle button to draw shapes.	Circles or rectangles are shown on the whiteboard.	Pass
28	Eraser	Users can erase anything on the whiteboard.	Users click eraser button and drag mouse to erase content on the whiteboard.	The path eraser passes by will be covered in white trail different from the canvas color.	Pass
29	Upload	Users can upload pdf to the whiteboard.	Users right click the pdf button to upload pdf.	Pdf file is shown on whiteboard.	Pass
30		Users can upload image to the whiteboard.	Users right click the image button to upload image.	Image file is shown on whiteboard.	Pass

Test ID	Test Classification	Test Description	Execution Step	Expected Output	Pass / Fail
31		The last uploaded file can be moved to anywhere on the whiteboard.	Users move the file.	The file is moved to specific place.	Pass
32		Every uploaded file can be moved to anywhere on the whiteboard.	Users move every file.	Every file can be moved to specific place.	Fail
33		The last uploaded file can be zoomed or shrink.	Users zoom or shrink the file.	The file is zoomed or shrink.	Pass
34		Every uploaded file can be moved to anywhere on the whiteboard.	Users zoom or shrink every file.	Every file is zoomed or shrink.	Fail
35	Chat	Users can send messages through enter key.	Users can send messages through enter key.	Messages can be sent.	Pass
36		Multiple users can send messages through send button.	Multiple users can send messages through send button.	Messages can be sent.	Pass
37		Single user can send messages through send button.	Singel users can send messages through send button.	Messages can be sent.	Fail

Test ID	Test Classification	Test Description	Execution Step	Expected Output	Pass / Fail
38		Users can send emoji.	Users choose emoji to send.	Emoji is sent.	Pass
39		Previous messages can be seen by new users and offline old users.	New users and offline old users enter the room.	Previous messages can be seen by new users and offline old users.	Fail
40		Every sent message display sender's name.	Users send messages.	User name is displayed with the sent message.	Pass
41		Every sent message display time stamp.	Users send messages.	Time stamp is displayed with the sent message.	Fail
42		Files of any type can be shared.	Users click shared file button to share files.	File is shared and other users can download it.	Pass
43	Undo	Users can undo their drawings on the whiteboard.	Users click undo button to undo the drawings.	Drawings are undone.	Pass
44		Users can undo the insertion including images and shapes.	Users click undo button to undo the insertion	Insertions are undone.	Pass
45		Users can undo the movement of images or shapes.	Users click undo button to undo the movement.	Movement is undone.	Pass

Test ID	Test Classification	Test Description	Execution Step	Expected Output	Pass / Fail
46	Video/Audio	Users can join the room with audio and video.	Users join the room and agree the permission of camera and microphone.	Users join the room with video and audio.	Pass
47		Users can join the room with either video or audio.	Users join the room and deny the permission of camera or microphone.	Users join the room with either video or audio.	Fail
48		Users can join the room without audio and video.	Users join the room and deny the permission of camera and microphone.	Users join the room without video or audio.	Fail
49		Users can adjust room creator's volume.	Users adjust room creator's volume.	User can mute room creator or adjust room creator's volume.	Pass
50		Users can adjust other users' volume.	Users can adjust other users' volume.	User can mute others or adjust others' volume.	Fail

Test ID	Test Classification	Test Description	Execution Step	Expected Output	Pass / Fail
51	LaTeX	Users can generate LaTeX formula and put it in the whiteboard.	Users write LaTeX codes and press render button to generate image and store it in the local and import the image.	LaTeX formula is displaced in the whiteboard as image.	Pass
52		Users can move or zoom or shrink LaTeX formula.	The same as image	The same as image.	Pass
53	Compatible browsers	Whiteboard can be used in Chrome, Firefox, Safari, IE, Microsoft Edge.	Users use one of the browsers to link to the whiteboard.	Whiteboard can be implemented on all mentioned browsers.	Pass

Appendix D

User Manual

Purpose:

The manual is for the user of the shared whiteboard software developed by GRP team 13. It will give the user a quick guide of what the whiteboard can do and how to use it.

Main functionalities:

Support for multi-users collaborative operations.

Support the writing of pencil and marker.

Support eraser by white color pen.

Support for varies of graphical inputs, like circle, rectangle, arrows, Bezier curve.

Provides video and audio communications at the same time.

Provides a text chatting box.

Can render LaTeX code into formula images.

Environment requirements:

Operating system: Windows, Linux, MacOS and also iOS.

Network: different users should be in same local area network Browser:Edge, chrome, Firefox is recommended.

Hardware: Need the authority of camera and microphone.

How to deploy server

If you have your own web server and you want to run this whiteboard on it, you can follow below steps.

1. To run this application server you need to install node.js (v10.0 or superior) and also install npm.

2. All the required packaged were installed in node_modules

3. Then you can run 'node server –ssl' in the root folder.

Now you can open https://localhost:9001/ to open the home page.

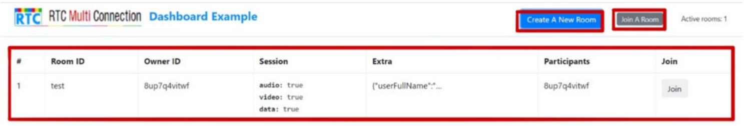
At present, this application is only allowed to be used in the same LAN.

Other users can access the application through the IP of the server and port 9001.

It is worth mentioning that if you are using Linux environment to deploy the project, when you do the 3rd step, you should modify the file "RTCMultiConnection-Server" into "rtcmulticonnection-server" otherwise an error occurs. In some other environment like MacOS system the problem does not exist, both capital and small letter file name can be used.

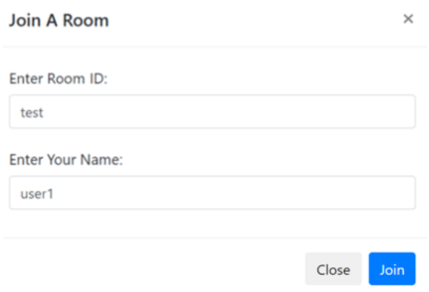
Instructions:

1. Start the main interface
   At the middle of screen users can see a table of all rooms have been created and the relevant information. User can choose to Create A New Room or Join A Room.
    By clicking on Join A Room, the first window pops up and requires the Room ID and the user ID, if the room exists, user can join it successfully. By clicking Create A New Room, the second window shows up, likewise it requires a new Room ID and user's name, optionally user can set the room password and the visibility of it.



#	Room ID	Owner ID	Session	Extra	Participants	Join
1	test	8up7q4vitwf	audio: true video: true data: true	{"userFullName":"...	8up7q4vitwf	Join

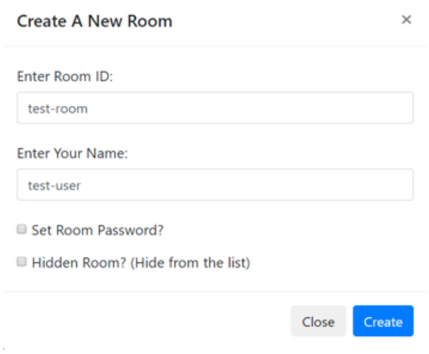
Figure D.1: *Home page demo*



Enter Room ID: test

Enter Your Name: user1

Figure D.2: *Room joining window*



Enter Room ID: test-room

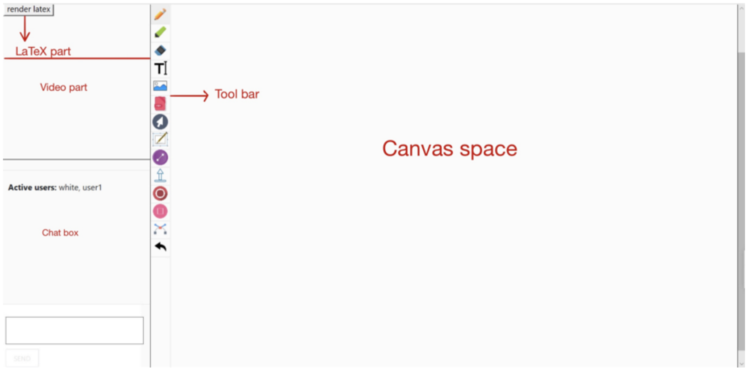
Enter Your Name: test-user

□ Set Room Password?

□ Hidden Room? (Hide from the list)

Figure D.3: *Room creating window*

2. After creating or joining, users are in their room page like the figure shown below.



Active users: white, user1

Figure D.4: *Room creating window*

There are five main parts as following:
Tool bar:
Users can choose different tools from this bar, such as pencil, marker and eraser; Also, users can insert texts, arrows, lines, images, pdf documents, rectangles, circles into the whiteboard.

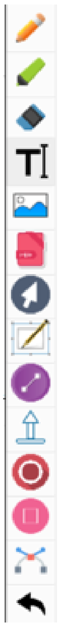


Figure D.5: *tool bars of the canvas*

Pencil: pencil to draw paintings
Marker: thicker but lighter pencil.
Eraser: Use a white pen to erase the marks.
Write text: Click anywhere in the canvas to insert the text object.
Add image: Click to insert images into the canvas.
Add pdf: Click to insert pdf images into the canvas.
Drag/move last path: Move the last stroke.
Drag/move all paths: Move all paintings in the canvas together.
Draw lines; Draw Arrows; Arc; Rectangle; Bezier curve: To draw those

shapes

Video/audio panel:

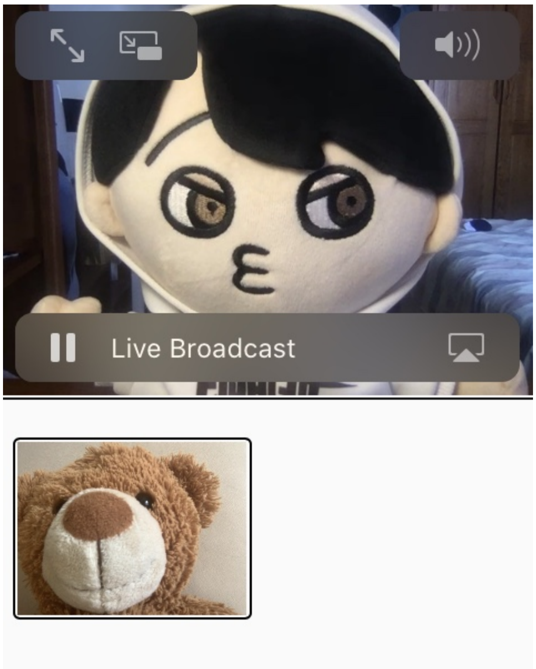
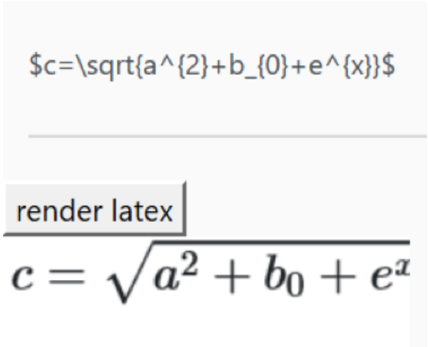


Figure D.6: *Video panel*

To enter the room, the authority of camera and microphone is required, once you are in the room, the video/audio communications is connected automatically, the video images of all members of the meeting is displayed on the left side. You can pause the video by clicking on the pause button, and mute others or yourself by clicking mute button. All active users will be displayed under the video image.

LaTeX tool:



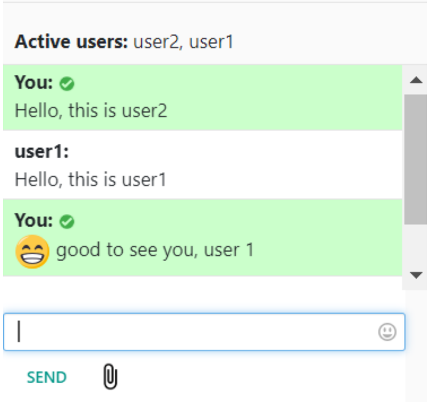
```
$c=\sqrt{a^{2}+b_{0}+e^{x}}$
```

$$c = \sqrt{a^2 + b_0 + e^x}$$

Figure D.7: *LaTeX formula rendering demo*

User can render the LaTeX notation into formula image by the LaTeX tool on the top left corner. Type in the LaTeX-format source code and click 'render' button, the corresponding formula image will be displayed above the button. After this, users can save the image on local machine and then insert the formula as a picture into the whiteboard.

Chat room:



Active users: user2, user1

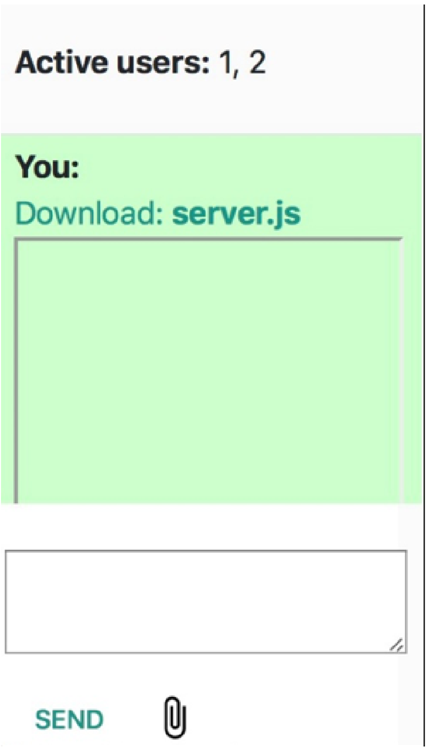
You: ✓
Hello, this is user2

user1:
Hello, this is user1

You: ✓
😁 good to see you, user 1

Figure D.8: *chatting box demo between two users*

A chatting box for text communication is provided on the bottom-left corner. Users can type in the message in the dialog box and click 'SEND' to send it, the context should be no longer than 120 characters.



Active users: 1, 2

You:
Download: **server.js**

Figure D.9: *chatting box demo between two users*

A file sharing demo The chatting room also provides file sharing function. By click the clip next to SEND button, you can share your document from your own compute for other users to download.

Appendix E

Meeting Minutes

1. Week One

   Date: 2019.10.18
   Time: 15:00 – 16:00
   Venue: Ornamental Garden
   Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei
   Late: None
   Absence: YIN Tianzhu
   Chairperson: None
   Minutes Taker: DANG Wenfei

   Content:
   1. Meeting Minutes
   Writing meeting minutes is necessary and all team member should write GRP meeting minutes in turn every week.
   2. Main Requirements for Shared WhiteBoard
   1) Easy to use
   2) Easy to install
   - Can be installed on any current computer platform
   3) Easy to update
   4) Be free to use
   5) Be high quality to use
   6) Be responsive
   - Avoid message transmission delay
   - Important than accuracy
   7) Support drawing by digital pen
   8) Support type in formulas in notation
   - Pop up a dialog window
   - Allow user to type in formula in this window
   - Allow user to drag the result output to anyplace they want on screen
   9) Support video call
   10) Support audio call
   11) Use cloud-based software
   - Run programs on local machine

- Save the states on cloud

3. Steps Need to Do Next

1) Think as users

- What functions do users want or need

2) Settle the assignments

- Distribute assignments to each of team member
- Be suitable for team members

3) Learn form current white board system

- What function can be reused
- What function is missing

4) Think all libraries and important components the software needed

5) Put all the libraries and components in program

2. Week Two

Date: 2019.10.25
Time: 14:00 – 15:00
Venue: TB 221
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei, YIN Tianzhu
Late: None
Absence: None
Chairperson: None
Minutes Taker: XIA Qing

Content:
1. Report on What We Did Last Week -the user/system requirement -the project website http://cslinux.nottingham.edu.cn/~Team201913/index.html

2. Our Problem:
1) login system -Zoom can be seen as a reference -one to one can share the link or name -can keep login system optional

2) avoid input crash
-set the presenter during the conversation
-when many people use the white board together manage who the presenter is
-concurrency problem for instance two people want to write the same area
-presenter is given the only person who can write
-other people can write until the presenter release it
-using layer maybe have some problem with undo operation need to have a record system
-let the white board look like a piece of paper

3) delay problem
-away distance can cause delay
-click on the button same time
-allow people write same time but may not know who did first accurately

4) problem of library searching -search professional components with open source system -specific library in specific system

5) platform - allow for all platform - laptop computer PC system
- Java is considered goo choose

6) web-based application
-peer to peer
-Connect through completed graph system allow every participant to communicate to others

-more security
-data sharing more efficient
-faster than download
-save the data on machine that means people need to use the same machine next time
-centralist server
-allow user to communicate through the server
-user need account (log in system need server)
-can see history from other account
-the number of users with multiple machine is a problem

7) GUI (Graphical User interface)
-link all components together but not do anything
-Java can do graphics

3. Steps Need to Do Next Week
-think about component
-look for library which is free and open source level
-Think about how to connect them

4. Others: our project has no ethical problem
-the first semester designing the basic system
-the second semester just test and improve the system

3. Week Three

Date: 2019.11.1
Time: 14:00 – 15:00
Venue: TB 221
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei, YIN Tianzhu
Late: None
Absence: None
Chairperson: None
Minutes Taker: LUO Chengtao

Content:
1 report on what we did last week
1.1 show the library including Yunxing, shared whiteboard of matisse in GitHub. We got some problem on showing the YunXing demo because we did not pre-run it before meeting.
1.2 ask for answer for some question
1.2.1 code plagiarism: we submit the source without any change, that is plagiarism. Once we add some features, it is not.
1.2.2 we cannot run the matisse's shared whiteboard on local laptop: the program is decided for linux.
1.2.3 time change for next week's meeting: from 2-3 pm Friday to 3-4 pm Thursday

2 license of software we find it is divided in paid, free, open source and so on. We need the software completely open source and allow us to do whatever we want.

3 some might-be problem of our library the limitation (class) of the demo, is it for personal development or enterprise for each library we find so far, does it match all requirement? If not, what does it miss? we should find another one to fulfill it.

What we need to do for the next week
1. design a UI
2. find the license of the library. If it does not give us enough permission, we find another one.
3. improve the user requirement and system requirement.
4. initiate the interim report

4. Week Four

Date: 2019.11.7
Time: 15:00 – 16:00
Venue: SEB 449
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei, YIN Tianzhu
Late: None
Absence: None
Chairperson: None
Minutes Taker: Naixuan HE

Content:
1 report on what we did last week
1.1 show the up-dated user requirement and a prototype of Whiteboard
1.2 explain the general idea of our software
1.3 show the general time plan

2 wonder about what to do next
It's better to start programming the Whiteboard step by step. First, realize a simple/basic function of the app, for example: texting, and then add other functions and features into it.

3 Where to deploy the Whiteboard
For now, use the school server to realize basic functions. In the future, if the school server cannot meet our requirement, we can find a new server. Joseph Zhang may provide some advice about it then.

4 The current whiteboard library cannot be deployed
The current whiteboard demo is too old and can only be deployed on a lower version of environments. It is better to find a more recent whiteboard library rather than insist on using the current one.

What we need to do for the next week
1. Find a license-ok new whiteboard library
2. Try to realize one basic function of whiteboard so that it is available to demonstrate in the next meeting (hope to realize it in one-week-time)

5. Week Five

   Date: 2019.11.14
   Time: 15:00-16:00
   Venue: SEB-449
   Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei, YIN Tianzhu
   Late: None Absence: None
   Chairperson: None
   Minutes Taker: BAI Yufei

   Content:
   Q1: Deploy the server program on Linux. But on which machine?
   Ask Joseph: to deploy the server on a Dell workstation which Amin bought.
   Q2: what to do next?
   -Advice: Make it more colorful.
   -Adding features/functionalities.
   -Add functionality: to see what others draw. (talk between devices.)
   -How the server take care of it, how long it's gonna take.
   Q3: Can implement different protocols at same time.
   Q4: OK for now if the app is only available for chrome. But in the future its preferred to support multiple browsers.
   Q5: OK to use SDK to develop our own project.

6. Week Six

Date: Nov 21st, 2019
Time: 15:00-16:00 (GMT+8)
Place: PMB449
Chair: Amin
Secretary: Tianzhu Yin
Present: Tianzhu Yin, Chengtao Luo, Naixuan He, Wenfei Dang, Qing Xia, Yufei Bai

Main Discussion:
i. System Requirement specification
ii. Problems and researches of Audio/video
Audio/video demo can only be used on LAN, not on Internet, and it is unstable.
iii. Decide to use WebRTC as the communication protocol
iv. Amin explained what firewall and NAT is. Firewall is a technology that helps a network build a isolated protective gate between its internal and external networks to protect data and information security.
NAT is short of network address translation, a software which used for when some hosts in the private network have been assigned to the local IP address (the private address only used in the private network), but now they want to communicate with hosts on the Internet (without encryption).
v. Connection with Amin's server Files deployed on the server, and chatting room is available while audio/video is unavailable. We will deal with that further.

Next plan:
i. Write midterm report with LaTeX (Real-time collaboration)
ii. Improve audio/video part and make it available.

7. Week Seven

   Date: 2019.11.28
   Time: 15:00   16:00
   Venue: PMB 449
   Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, BAI Yufei, YIN Tianzhu
   Late: None
   Absence: HE Naixuan
   Minutes Taker: DANG Wenfei

   Content:
   1. Meeting Minutes
   Writing meeting minutes is necessary and all team member should write GRP meeting minutes in turn every week.

   2. Feedback of Recent Task
   1) Whiteboard (Heroku)
   - A good sample contains everything the program needed but LaTeX
   - UI design need to be modified
   - Can not create without the camera permission
   - Can not save the state of whiteboard
   - Resolution of the whiteboard is low
   - Room is delated when the room creator leaves

   3. Two Options
   1) Take the Heroku as the basement and add LaTeX on it.
   2) Develop our own program

   4. Task
   1) Decide which option to develop the program If take Heroku
   i. Improve the resolution
   ii. Find a way to add the functionality
   iii. Learn to use Heroku, try to deploy it on UNNC's server
   iv. Check licenses
   If develop our own
   i. Decide server vs. P2P (when is it necessary to have a server)

8. Week Eight
Date: 2019.12.5
Time: 15:00 – 16:00
Venue: TB 101
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei, YIN Tianzhu
Late: LUO Chengtao, XIA Qing
Absence: None
Chairperson: None
Minutes Taker: XIA Qing

Content:
1. Internal Report Problems
- Keep the format consistent
- Lack of cover and section number
- Lack of cohesion and unclear structure
- Introduction is unclear

2. Suggestion
Front Page
- Tittle -Authors [in alphabetic order of surname]
- Data
Part I
- Introduction (abstract)
What is shared whiteboard and what we want to do
- Background, Motivation -related work
search existing system and explain the strength and weaknesses
- requirements (what component is used) put in one table
- Technical background/preliminaries (explain why choose to use it) (consider the order problem)
Part II
- contribution: what we have done
- implementation details
Appendix
Challenges
Minutes
Gantt chart
3. Other Tips
Consider the Structure
-Part Chapter Section Subsection

9. Week Nine

Date: 2019.12.10
Time: 17:00 – 18:00
Venue: TB 220
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei, YIN Tianzhu
Late: None
Absence: None
Chairperson: None
Minutes Taker: LUO Chengtao

Content:
1. Internal Report Problems
- The format is not consistent
- Some spelling problem
- Academic problem
2. Suggestion
1) Need to contain
-Abstract
-Introduction
Introduction can be written more detailed with showing pictures.
-Conclusion
Conclude what we have done and what we have learned so far.
2) Technical issue part
-Just write what is related to the project, what we can realize
-NAT part
No need to write too much about it, just mention we find a solution for the server
problem but not promise to solve, just a little writing about it.
3)Picture
-We should show the pictures of working whiteboard or working component, a blank one may not be good.
-Pictures can be put in abstract part to show what is a shared whiteboard, give reader a clear explanation
3. Other Tips
1)Time plan: if we miss the last 5 percent work, we get nothing.
2)index about key word can be put in the end.

10. Tenth Minutes

Date: 2019.12.19
Time: 14:00 – 15:00
Venue: PMB 449
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, BAI Yufei, YIN Tianzhu
Late: None
Absence: HE Naixuan
Chairperson: None
Minutes Taker: LUO Chengtao

Content:
1. What we have achieved
- Show the whiteboard component which is from:
https://github.com/lovasoa/whitebophir
Show how it works and how it shared
2. Problem
a) Cannot contain kchat with the whiteboard
-solution
Focusing on LaTeX problem first.
b) Still don't know how to add LaTeX in whiteboard
-Solution
-implement operation inside the whiteboard, which means that writing code, transfer it into svg and add it to whiteboard these three operations are all done within whiteboard.
-use an extension which can be learned from quickTeX, writing code and transfer it into svg, and then store it in the local space and upload it to whiteboard.
3. For next week Focusing on Latex problem, try the two methods mentioned above

11. Eleventh Minutes

Data:2020.3.5
Time: 16:00 – 17:00
Venue: Online through Zoom and WeChat
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei, YIN Tianzhu
Late: None
Absence: None Chairperson: None
Minutes Taker: Naixuan HE

Content:
1.What we have achieved
-Render LaTeX equation, transfer LaTeX code to svg format.

2. Problem
1)Failure to display the svg equation maybe because of the transmission.
-Solution
Find other whiteboards to have a try.
2) Work distribution is unbalanced among team members, only 2 are working.
-Solution
2 members work on code, the others work on final report.

3. For next week
1)Plan
Try all possibilities and see if we can make code progress

12. Twelfth Minutes

    Date: 2020.3.12
    Time: 16:00-17:00
    Venue: Online through Zoom and WeChat
    Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei
    Late: None
    Absence: YIN Tianzhu
    Chairperson: None
    Minutes Taker: He Naixuan

    Content:
    1.Final report
    1.1 Tools for writing the final report
    1.1.1 ButBucket(git-hub)
    VPN to register account
    Then use it freely without VPN
    1.1.2 Email is fine
    Can start now
    No need to wait for the functions to set up
    Take some material from intern report
    Share progress during every meeting (bitbucket etc.)
    2.Current Problem
    cannot combine LaTeX to WB
    solution: LaTeX-¿ Compiled Output -¿ Image -¿ added to WB
    1.independent to the WB (multiple ways)
    2.need to be figured out
    Current situation: can add pictures but cannot change location or size
    Can only locally applied
    -device under same Wi-Fi
    example:
    192.168.1.102
    192.168.1.103
    -outside is much more difficult
    -try to apply it under university network (school VPN)

    3.QA
    Can shared WB -¿ local WB?
    -Shared WB is preferred

13. Thirteenth Minutes

Date: 2020.3.26
Time: 16:00 – 17:00
Venue: Zoom WeChat
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, YIN Tianzhu, BAI Yufei
Late: None
Absence: HE Naixuan
Chairperson: None
Minutes Taker: DANG Wenfei

Content:
1. Final Report
Problem:
1. There are few contents in interim report could be used in final report
2. The program is not finished yet; final report relies a lot on the settled system
Solution:
Start with the relevant background, technology and introduction.
2. Results from Last Week
1) Chengtao's Whiteboard
a) Advantage: easy to render LaTeX formula b) Disadvantage: the formula is just showed on a side of the interface, cannot add it into the whiteboard
2) Qing's Whiteboard
a) Advantage: the rendered LaTeX formula can be added into the whiteboard, and user is able to draw things on that formula
b) Disadvantage: the process of adding formula is complex, as the rendering result is a picture, user need to save the picture on local machine first, then inset that into the whiteboard
3) Supervisor's comment
Both are great, keep them alive and try to figure whether one of them could work well between two users.
3. Steps Need to Do Next
1) Testing the whiteboard
a) With supervisor next Monday or Tuesday
b) With team members by using VPN, VDI or X2GO to connect campus network
2) Start the final report

14 Fourteenth Minutes

Date: 2020/4/2
Time: 16:00 – 17:00
Meeting Place: Zoom WeChat
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, HE Naixuan, BAI Yufei
Late: None
Absence: Yin Tianzhu
Chairperson: None
Minutes Taker: Yufei Bai

Q1: Where we deploy the program.
Advice: for now, at least try on 2 laptops
Can be done already.
Q2: How to do the presentation?
It's a recorded video.
Ricky shows the program on his two computers.
Q3: How the LaTeX formula is rendered?
Qing shows the latex work so far.
Advice: Make the procedure more efficient. Is there a way to make it automatic?
To render it to a file.png and put it on the whiteboard.
Q4: Why it doesn't work on the workstation?
The problem is that when a room is created, you can't connect to each other, people are not actually in the same room. May arrange a meeting to solve the problem.
For last week, we wrote something about the abstract, background and the tech we used.
In this week: Try to add more features on the program and try to make progress on writing the report, maybe send a demo version to Amin next week.

15. Fifteenth Minutes

Date: 2020/4/9
Time: 16:00-17:00 Venue: Zoom
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, YIN Tianzhu, He Naixuan, Bai yufei
Late: Yin Tianzhu
Minutes Taker: XIA Qing

Content:
1.Update system design
Change and develop
Write the stuff we have now
Have a system with the whiteboard and the latex functionality how to design it and how them work
2.Assume that other people know nothing with the system Report include all of specification of the system
What is the whiteboard
What is the latex functionality and how to add it 3.What we will intend to do in the future
All the problem (from the library, from the license, a little bit to work the design and ideal design)
How the current one works
How we work from week 1 to now
4.Design of the system
How you make different component link together
How the variable component connect to each other
Foundation / basic add other stuff
What language use
5.Latex code -¿ through a compiler (Mathjax engine) -¿ To a png -¿put that onto the white-broad
How to Integrate them Can include UML Do not to discussion the thing that we not do for our project in the foundation. Justify our design

Step Need to Do Next
Draft of report before next Wednesday
Do not leave to the last minutes

16. Sixteenth Minutes

Date: 2020.4.16
Time: 16:00 – 17:00
Venue: Online through zoom
Attendance: LUO Chengtao, XIA Qing, DANG Wenfei, BAI Yufei, HE Naixuan
Late: None
Absence: YIN Tianzhu
Chairperson: None
Minutes Taker: LUO Chengtao

Content:
1. What we have achieved
- achieve final whiteboard and final report draft

2. Question
1) In the hand book, it says implementation of the system of the components. Here, the component means LaTeX functionality? Do we need to write others?
-Answer
How LaTeX functionality interacts with foundation whiteboard, the procedure of how LaTeX works
Just introduce where the open source comes from, no need to explain how it works
2) For the last meeting, you said updated design is about how LaTeX links to the foundation whiteboard, including failure and success tries. I think this is quite similar to the part of implementation of the LaTeX
-Answer
Implementation part is about final product, and updated design includes both failure and success try (no need to write it here in detail, the detailed part is in implementation)
3) We don't understand the meaning of the overview of developed source code hierarchy and we don't know how to write.
-Answer
It is about how written codes link to each other, and some important code can be attached so that readers can easily know what we have done.

3. Feedback of the draft
1) In UI part, images are not good, need changing.
2) Abstract, introduction, summary are the three most important parts, need to be good.
3) Images or diagrams can be added, but not too many.
4) Two main diagrams show the relationship of components
Diagram 1: the interaction of components in whiteboard
Diagram 2: the interaction of LaTeX
5) Future Plan
How we will improve the whiteboard

Need a specific section to write it
6) Improve reference part

4. For Future if ddl is delayed
Improve LaTeX functionality, make it more automatically.