```
class Chatbox {
    debugger;
    constructor() {
        this.args = {
            openButton: document.querySelector('.chatbox__button'),
            chatBox: document.querySelector('.chatbox__support'),
            sendButton: document.querySelector('.send__button')
        }

        this.state = false;
        this.messages = [];

    }

    display() {
        debugger;
        const {openButton, chatBox, sendButton} = this.args;
        openButton.addEventListener('click', () => this.toggleState(chatBox))
        sendButton.addEventListener('click', () => this.onSendButton(chatBox))

        const node = chatBox.querySelector('input');
        node.addEventListener("keyup", ({key}) => {
            if (key === "Enter") {
                debugger;
                this.onSendButton(chatBox)
                $("#wave").css("display", "block");
            }
        })
    }

    toggleState(chatbox) {
        this.state = !this.state;

        // show or hides the box
        if(this.state) {
            debugger;
            chatbox.classList.add('chatbox--active')
        } else {
            debugger;
            chatbox.classList.remove('chatbox--active')
        }
    }

    onSendButton(chatbox) {
        debugger;
        var textField = chatbox.querySelector('input');
        var jsonObject = [{"recipient_id":"Bot", "text":"Error while retrieving"}]
        let text1 = textField.value
        textField.value = ''
        if (text1 === "") {
            return;
        }

        let msg1 = { name: "User", message: text1 }
        this.messages.push(msg1);
```

```
        // this.updateChatText(chatbox)

        $.ajax({
            url:'http://localhost:8000/chatbot/botresponse/',
            type: 'POST',
            dataType: "json",
            // cache: false,
            // processData: false,
            async: false,
            mode: 'no-cors',
            CSRF_TRUSTED_ORIGINS : ['http://*', 'https://*'],
            contentType: "application/json; charset=utf-8",
            data : JSON.stringify({
                "message": text1,
                "sender": "Bot"
                }),
            success: function (response) {
                jsonObject = JSON.parse(JSON.stringify(response))
            },
            error: function (response) {
                    console.log("Error");
                }
            });
            if (jsonObject != null) {
                for (let i = 0; i < jsonObject.length; i++) {
                  let msg = { name: jsonObject[i].recipient_id, message:
jsonObject[i].text,  };
                  if (jsonObject[i].buttons && jsonObject[i].buttons.length > 0) {
                    msg.buttons = jsonObject[i].buttons;
                    console.log("contains buttons");
                    console.log(msg.buttons);
                }
                  this.messages.push(msg);
                }
                this.updateChatText(chatbox);
              }


        }

    // updateChatText(chatbox) {
    //     var html = '<div class="messages__item messages__item--operator">' +
'mistake' + '</div>'
    //     const chatmessage = chatbox.querySelector('.chatbox__messages');
    //     chatmessage.innerHTML = html;
    // }


    updateChatText(chatbox) {
        debugger;
        var html = '';
        this.messages.slice().reverse().forEach(function(item, index) {
            debugger;
            if (item.name === "Bot")
            {
```

```
                if (item.buttons && item.buttons.length > 0) {
                    // display item.buttons in the chatbox as buttons
                    html += '<div class="messages__item messages__item--visitor">';
                    item.buttons.forEach(function(button) {
                        const buttonElement = `<button class="button"
onclick="bot_button('${button.title}')">${button.title}</button>`;
                        html += buttonElement;
                    });
                    html += '</div>';
                }
                html += '<div class="messages__item messages__item--visitor">' +
item.message + '</div>'
            }
            else
            {
                html += '<div class="messages__item messages__item--operator">' +
item.message + '</div>'
            }
        });


        debugger;
        const chatmessage = chatbox.querySelector('.chatbox__messages');
        chatmessage.innerHTML = html;
        // $("#wave").css("display", "none");

    }
}



const chatbox = new Chatbox();
chatbox.display();


function bot_button(x){
    const chatbox = new Chatbox();
    // need to send x to onSendButton
    html = '<div class="messages__item messages__item--operator">' + x + '</div>'
    alert(x, "nitin")
}


function chatfunction(x){
    alert(x.val);
}


function on_microphone() {
    var recognition = new webkitSpeechRecognition(); // Create a new instance of the
SpeechRecognition object

    document.getElementById("microphone-button").addEventListener("click", function() {
    recognition.start(); // Start speech recognition when the microphone button is
clicked
```

```
    });

    recognition.onresult = function(event) {
    var result = event.results[0][0].transcript; // Get the transcribed speech
    document.getElementById("search-input").value = result; // Set the transcribed
speech as the search input value
    recognition.stop(); // Stop speech recognition
    }
}
```