CODE: Github link

1) Frontend

```
08 🔲 🗎 🖽
                                                D □
Ф
     ∨ WEBSOCKET
     > 🗷 .vscode
         package.json
README.md
go
                                                       render.yaml
æ>
                                                                      margin: 0;
padding: 0;
box-sizing: border-box;
田
8
0
                                                                      padding: 20px;
color: ■#333;
                                                  20
21
22
                                                                      max-match: 3copts,
max-match: 3copts,
margin: 0 auto;
background: □rgba(255, 255, 255, 0.95);
backdrop-filter: blur(10px);
border-radius: 20px;
box-shadow: 0 20px 40px □rgba(0, 0, 0.1);
overflow: hidden;
animation: slideIn 0.6s ease-out;
                                                                 @keyframes slideIn {
   from {
(
                                                                         om {
  opacity: 0;
  transform: translateY(30px);
> OUTLINE > TIMELINE
```

Fig 1.1'

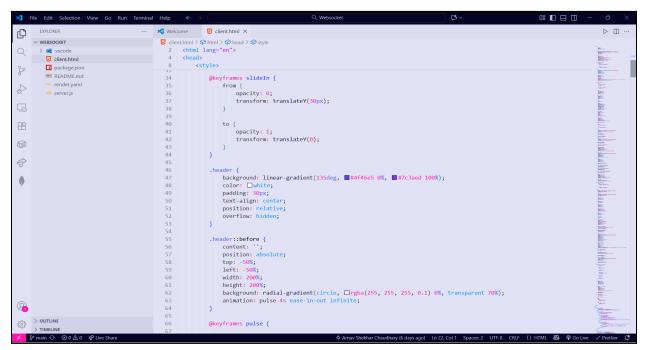


Fig 1.2

```
★ File Edit Selection View Go Run Terminal Help
                                                                                                                                                                                                                                                         0: 🛮 🗎 🗓
                                                                   ⊳ ⊞
D
        ∨ WEBSOCKET
                                                                    ☐ client.html > �� html > �� head > �� style
          > 💐 .vscode
                                                                                    <style>
             package.json
go
                                                                                        @keyframes pulse {
             README.md
              m render.yaml
æ>
                                                                                                 100% {
transform: scale(1);
opacity: 0.5;
胎
                                                                      73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
                                                                                                 50% {
    transform: scale(1.1);
    opacity: 0.8;
8
                                                                                           .header h1 {
	font-size: 2.2em;
	font-weight: 300;
	letter-spacing: 1px;
	position: relative;
	z-index: 1;
•
                                                                                          .header .subtitle {
   margin-top: 8px;
   opacity: 0.9;
   font-size: 1.1em;
   position: relative;
   z-index: 1;
(
                                                                                                padding: 30px;
> OUTLINE
```

Fig 1.3

2) Backend

```
08 🔲 🗎 🖽
                                                                                               😈 client.html 🔲 package.json 💌 render.yaml JS server.js
                                                                                                                                                                                                                                                                                                             D □
Ф
                                                                        JS serverjs >...
1  const WebSocket = __require('ws');
2  const http = require('http');
3  const fs = require('fs');
4  const path = require('path');
        ∨ WEBSOCKET
                                                ព្ធប្រ
         > 💐 .vscode
              nackage.json
go
              README.md
                                                                                     // Create HTTP server for serving static files
const server * http.createServer((req, res) => {
   if (req.url == '/' || req.url == '/index.html') {
    fs.readfile(path.join(_dirname, 'client.html'), (err, data) => {
<<u>√</u>
       JS server.js
                                                                                            if (err) {
    res.writeHead(404);
    res.end('Not found');
    return;
田
res.writeHead(200, { 'Content-Type': 'text/html' });
8
                                                                                               res.end(data);
                                                                                       res.end(data);
});
} else {
res.writeHead(404);
res.end('Not found');
 0
                                                                                    });
                                                                                    // Create WebSocket server
const wss = new WebSocket.Server({ server });
                                                                                     wss.on('connection', (ws, req) => {
  console.log('New client connected from:', req.socket.remoteAddress);
  clients.add(ws);
                                                                                        // Send welcome message
ws.send(JSON.stringify({
   type: 'welcome',
   message: 'Connected to WebSocket server',
(2)
> OUTLINE
                                                                                                                                                                                                                 Ln 1, Col 1 Spaces: 2 UTF-8 CRLF {} JavaScript 👸 🖗 Go Live 🛷 Prett
```

Fig 2.1

```
★ File Edit Selection View Go Run Terminal Help
                                                                                                                                                                                                                                                                   0: 🛮 🗎 🗓
         EXPLORER
                                                                                             ⊳ ⊞
D
        ∨ WEBSOCKET
                                              ា្ធបគ្
                                                                       JS serverjs >...
30 wss.on('connection', (ws, req) ⇒ {
          > 💐 .vscode
                                                                                         timestamp: new Date().toISOString()
              package.json
go
              ■ README.md
                                                                                    // Handle incoming messages
ws.on('message', (data) => {
   try {
      const message = JSON.parse(data);
      console.log('Received:', message);
}
                                                                         41
JS server.js
               m render.yaml
                                                                                           // Broadcast message to all connected clients
const broadcastMessage = {
   type: 'broadcast',
   data: message,
   timestamp: new Date().toISOString(),
   clientCount: clients.size
}
出
                                                                         48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
8
 •
                                                                                            clients.forEach(client => {
   if (client.readyState === WebSocket.OPEN) {
      client.send(JSON.stringify(broadcastMessage));
   }
}
                                                                                        } catch (error) {
  console.error('Error parsing message:', error);
                                                                                     });
                                                                                    // Handle client disconnect
ws.on('close', () => {
  console.log('Client disconnected');
  clients.delete(ws);
}
                                                                                     });
@
                                                                                    // HandLe errors
ws.on('error', (error) => {
  console.error('WebSocket error:', error);
> OUTLINE > TIMELINE
    P main ⊕ ⊗ 0 🛆 0 🕏 Live Sh
```

Fig 2.2

```
o: 🛮 🗎 🗇
                                                           Ð
                                                                                                                                                                                                                                                          D □
       ∨ WEBSOCKET
                                        ា្ដ្រា
                                                                     wss.on('connection', (ws, req) => {
  wss.on('message', (data) => {
       > 💐 .vscode
                                                               42
                                                              61
62
63
                                                                              console.error('Error parsing message:', error);
go
           package.json
README.md
            m render.yaml
JS server.js
                                                                      // Handle client disconnect
ws.on('close', () => {
  console.log('Client disconnected');
  clients.delete(ws);
}
田
                                                                      // HandLe errors
ws.on('error', (error) => {
  console.error('WebSocket error:', error);
  clients.delete(ws);
8
                                                                         });
0
                                                                      const PORT = process.env.PORT || 3000;
server.listen(PORT, () => {
    console.log(`Server running on http://localhost:${PORT}`);
    console.log(`WebSocket server ready for connections`);
@
> outline
```

Fig 2.3

OUTPUT:

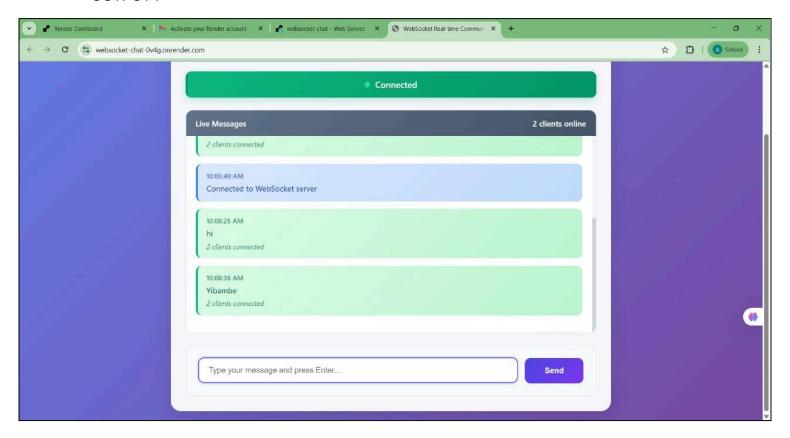


Fig 3.1 Real time message communication

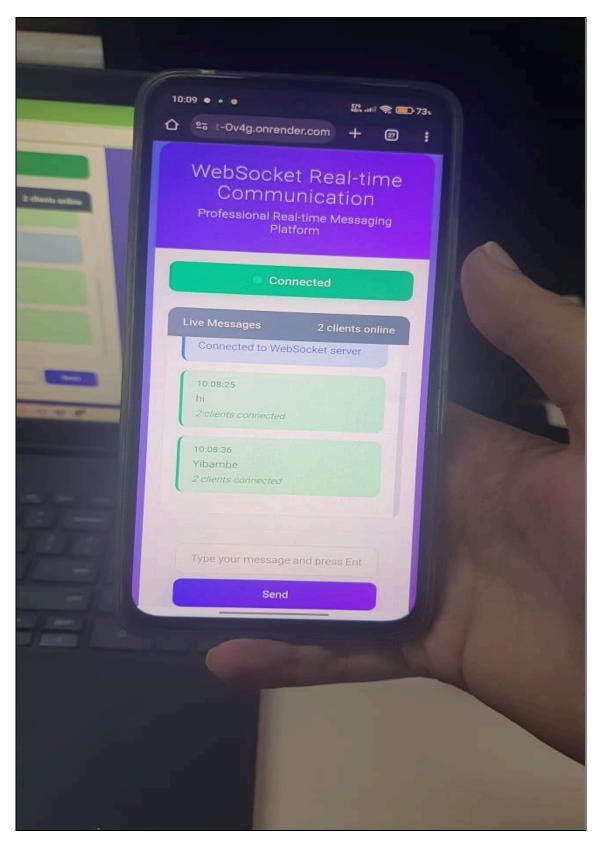


Fig 3.2: hosting of the software