

LAB 3 - SOCKET CHATT

Gruppmedlemmar:

Jonas Glingert

Lisa Pääjärvi

Tony Martinsson

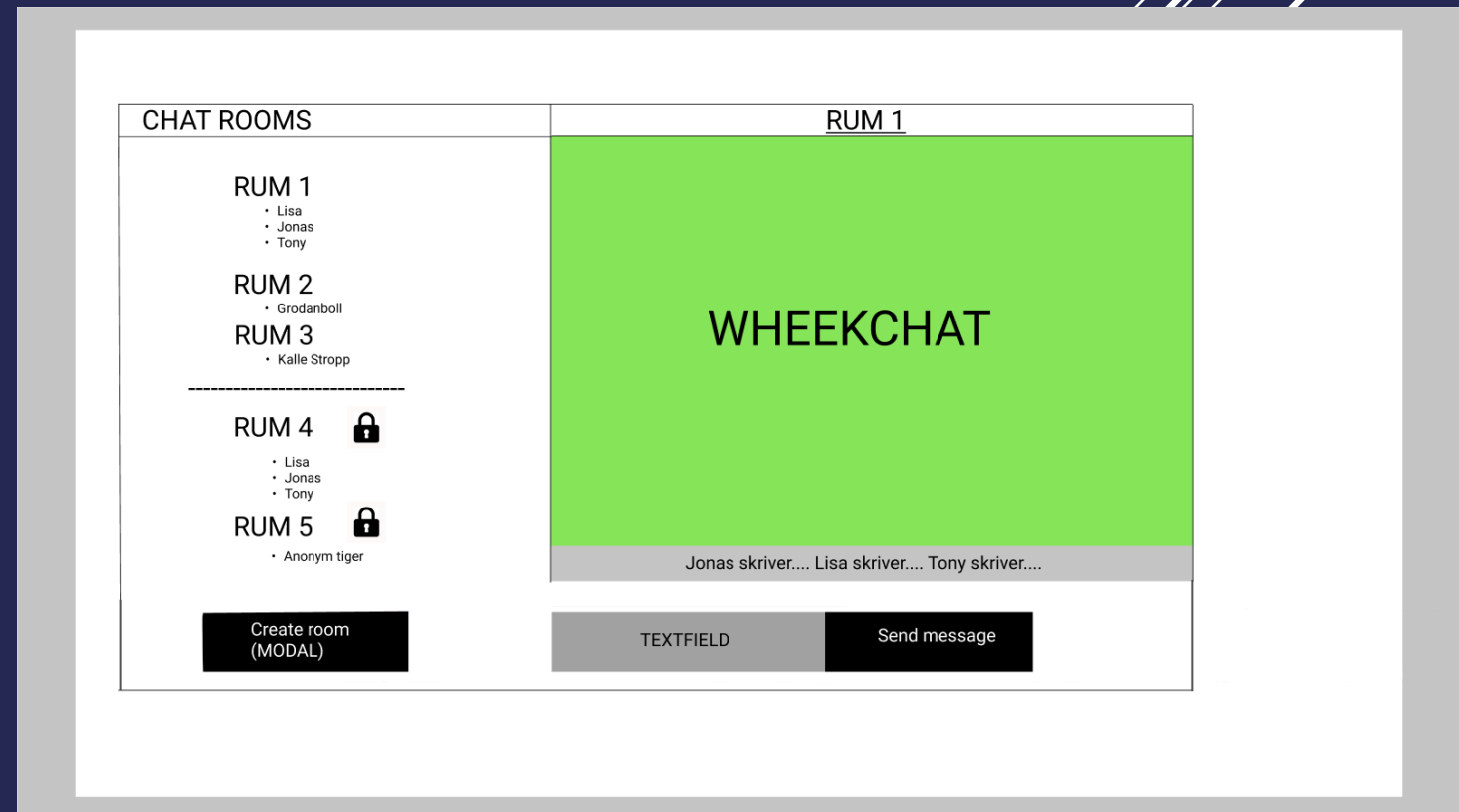
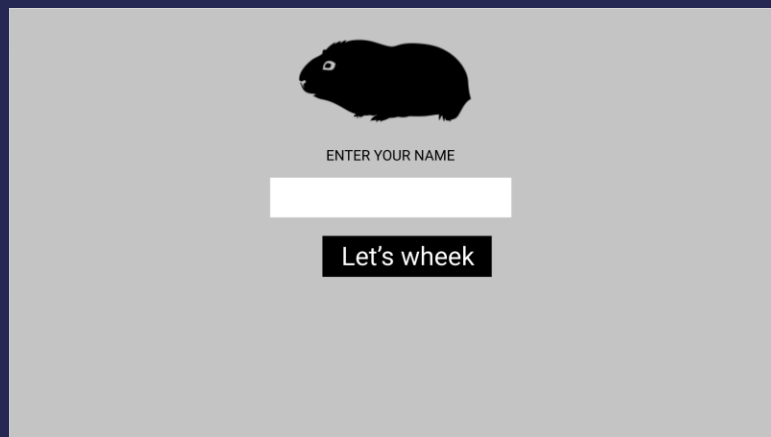
2021-05-12 FED20G



WHEEKCHAT

- * ENKEL TEXT-CHATT APPLIKATION
- * ANVÄNDER SIG AV SOCKET.IO
- * ANVÄNDARE KAN VÄLJA ETT ANVÄNDARNAMN, SKAPA ÖPPNA ELLER LÖSENORDSSKYDDADE RUM OCH CHATTA I REALTID

WIREFRAMES



DEPENDENCIES

SERVER:

SOCKET.IO

EXPRESS

NODEMON

CORS

CLIENT:

SOCKET.IO-CLIENT

MATERIAL UI

REACT

KODEXEMPEL

-CONTEXT

```
7
8 export function UserProvider({ children }) {
9   const ENDPOINT = 'http://127.0.0.1:4000/';
10  const [socket] = useState(io(ENDPOINT, { transports: ['websocket', 'polling'] }));
11  const [user, setUser] = useState();
12  const [messages, setMessages] = useState([]);
13  const [room, setRoom] = useState();
14  const [rooms, setRooms] = useState([]);
15
16  useEffect(() => {
17    socket.on('connect', () => {
18      console.log("Connected to server");
19      socket.emit("getRooms", {})
20    });
21
22    socket.on("rooms", activeRooms => {
23      setRooms(activeRooms);
24    })
25  })
```

KODEXEMPEL -CHATBOXEN

```
return (  
  <div className="connectedStyle">  
    {room ? (  
      <>  
        <div className="titleBoxChat">  
          <Typography variant="h5">{user} @ {room}</Typography>  
        </div>  
        <div className="chatBox">  
          <ul className="ulStyle">  
            <Typography variant="h6">  
              {messages.map((msg, index) => (  
                <li className="textBox" key={index}>  
                  {msg.user + ': ' + msg.text}</li>  
              ))}</Typography>  
            </ul>  
          </div>  
          <div className="sendNewTextBox">  
            <TextField  
              id="sendtext"
```

KODEXEMPEL

-CHATROOMS

```
return (  
  <div className="roomStyle">  
    <div className="titleBox">  
      <img className="logoRooms" style={{width: '15rem', margin: '1rem' }} src={logo} alt="Logo" />  
    </div>  
    <ul>  
      <Typography variant="h6">  
        {openRooms.map((room, index) => (  
          <li className="roomLink" style={{listStyle: 'none'}} key={index} onClick={() => joinOpenRoom(room)}>  
            {room.name}  
          </li>  
        ))}  
      </Typography>  
    </ul>  
    <Divider style={{background: '#9DA1C2'}} />  
    <ul>  
      <Typography variant="h6">  
        {closedRooms.map((room, index) => (  
          <li className="roomLink" style={{listStyle: 'none'}} key={index} onClick={() => joinLockedRoom(room)}>  
            {room.name}  
            <LockIcon style={{marginLeft: '0.5rem'}} />  
          </li>  
        ))}  
      </Typography>  
    </ul>  
  </div>  
)
```

REFLEKTIONER

- Turbofartsprojekt. Stressigt! ☹️
- Socket.io var utmanande. Vi försökte använda oss av de inbyggda funktionerna till en början...
- Vi återgick till våra egna states och använde socket.io som en ren transportör av data
- Bra kunskap att ha med sig. Kanske vi bygger kundsupport med chattfunktion i framtiden?
- Några marsvin kom i alla fall med 😊

LIVE DEMO

A series of several thin, white, parallel diagonal lines extending from the bottom right towards the top right of the image, creating a sense of motion or a dynamic background.