

Nama : Alhambra Ferdinando

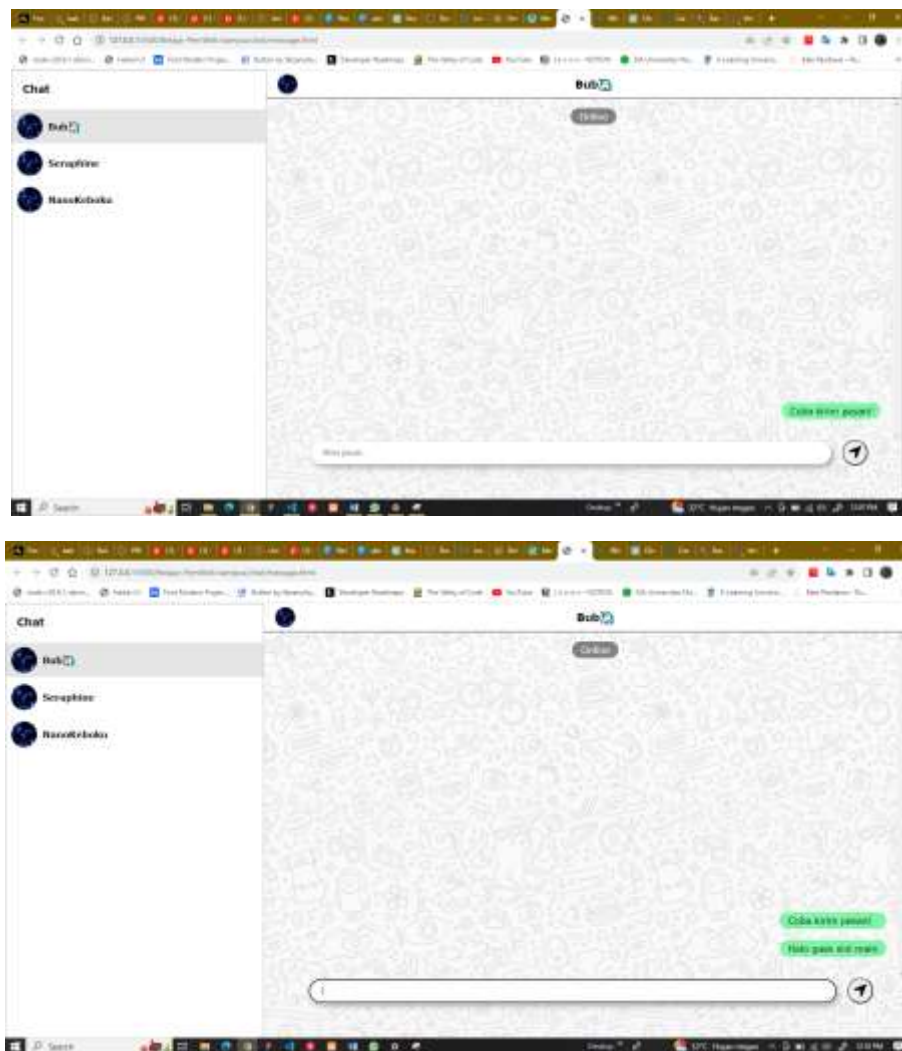
NIM : 222520090

Kelas C

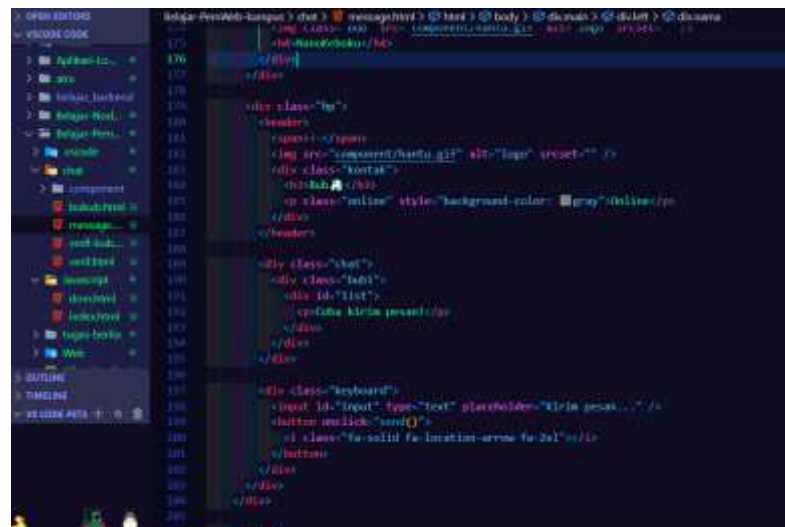
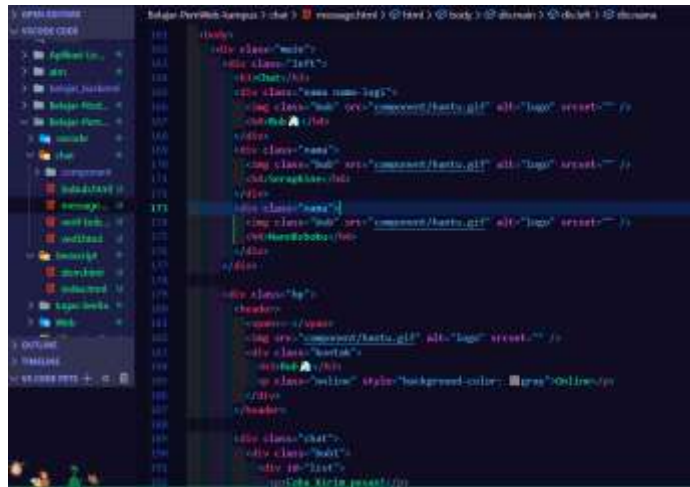
Semester 3

TUGAS PEMOGRAMAN WEB

Hasil codingan nomer 1 dan 2



Penjelasan :



- terdapat 2 side
- div class main sebagai pembungkus utama dari 2 slide
- div class left sebagai left-side yang menampilkan nama-nama kontak
- div class hp sebagai right side yang menampilkan room chat

- header sebagai nama kontak pada room chat
- div class chat untuk menampilkan pesan-pesan yang dikirim
- div class keyboard untuk menampilkan kolom chat berisi input type text dan button

The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays the file tree for a project named "Belajar PemWeb-kampus". The tree includes a "src" folder containing several HTML files ("bubuh.html", "message.html", "well-bub.html", "wellhtml.html") and a "javascript" folder containing "dom.html", "index.html", and "tugas-berita.html". There are also links to "OUTLINE", "TIMELINE", and "VS CODE FETS".

The main editor area on the right shows the code for "message.html". The code consists of HTML elements and a JavaScript function:

```
</button>
</div>
</div>
</div>

<script>
function send() {
    let input = document.getElementById("input");
    if (input.value == '') {
        alert("masukkan pesan!!");
        return false;
    } else {
        let node = document.createElement("p");
        let chat = document.getElementById("list");
        let textnode = document.createTextNode(`${input.value}`);
        node.appendChild(textnode);

        chat.appendChild(node);
    }
    input.value = '';
}
</script>

<script
src="https://kit.fontawesome.com/6bbf0bd217.js"
crossorigin="anonymous"
/>
</body>
</html>
```

Fungsi ini pertama-tama memeriksa apakah input kosong. Jika ya, ia menampilkan peringatan dan mengembalikan **false**. Jika tidak, akan membuat elemen pesan baru. kemudian menambahkan input pengguna ke elemen pesan dan menambahkan elemen pesan ke daftar pesan obrolan. Terakhir, membersihkan bidang input

Style Css

```

<div class="viewport" content="width=device-width,
<title>Message / TITIK
</title>
<style>
  * {
    margin: 0;
    padding: 0;
    text-decoration: none;
    list-style-type: none;
    border-collapse: collapse;
  }
  .main {
    display: flex;
    flex-direction: row;
  }
  .left {
    width: 40%;
    height: 300px;
    border: 1px solid #ccc;
    background-color: #fff;
  }
  .right {
    width: 60%;
    height: 300px;
    border: 1px solid #ccc;
    background-color: #fff;
  }
  .header {
    width: 100%;
    height: 50px;
    display: flex;
    padding: 5px 10px;
    align-items: center;
    justify-content: space-between;
    background-color: #fff;
  }
  .header span {
    position: relative;
    left: -20px;
  }

```

```

<div class="viewport" content="width=device-width,
<title>Message / TITIK
</title>
<style>
  * {
    margin: 0;
    padding: 0;
    text-decoration: none;
    list-style-type: none;
    border-collapse: collapse;
  }
  .main {
    display: flex;
    flex-direction: row;
  }
  .left {
    width: 40%;
    height: 300px;
    border: 1px solid #ccc;
    background-color: #fff;
  }
  .right {
    width: 60%;
    height: 300px;
    border: 1px solid #ccc;
    background-color: #fff;
  }
  .header {
    width: 100%;
    height: 50px;
    display: flex;
    padding: 5px 10px;
    align-items: center;
    justify-content: space-between;
    background-color: #fff;
  }
  .header span {
    position: relative;
    left: -20px;
  }

```

```

<div class="viewport" content="width=device-width,
<title>Message / TITIK
</title>
<style>
  * {
    margin: 0;
    padding: 0;
    text-decoration: none;
    list-style-type: none;
    border-collapse: collapse;
  }
  .main {
    display: flex;
    flex-direction: row;
  }
  .left {
    width: 40%;
    height: 300px;
    border: 1px solid #ccc;
    background-color: #fff;
  }
  .right {
    width: 60%;
    height: 300px;
    border: 1px solid #ccc;
    background-color: #fff;
  }
  .header {
    width: 100%;
    height: 50px;
    display: flex;
    padding: 5px 10px;
    align-items: center;
    justify-content: space-between;
    background-color: #fff;
  }
  .header span {
    position: relative;
    left: -20px;
  }

```

```

<div class="viewport" content="width=device-width,
<title>Message / TITIK
</title>
<style>
  * {
    margin: 0;
    padding: 0;
    text-decoration: none;
    list-style-type: none;
    border-collapse: collapse;
  }
  .main {
    display: flex;
    flex-direction: row;
  }
  .left {
    width: 40%;
    height: 300px;
    border: 1px solid #ccc;
    background-color: #fff;
  }
  .right {
    width: 60%;
    height: 300px;
    border: 1px solid #ccc;
    background-color: #fff;
  }
  .header {
    width: 100%;
    height: 50px;
    display: flex;
    padding: 5px 10px;
    align-items: center;
    justify-content: space-between;
    background-color: #fff;
  }
  .header span {
    position: relative;
    left: -20px;
  }

```

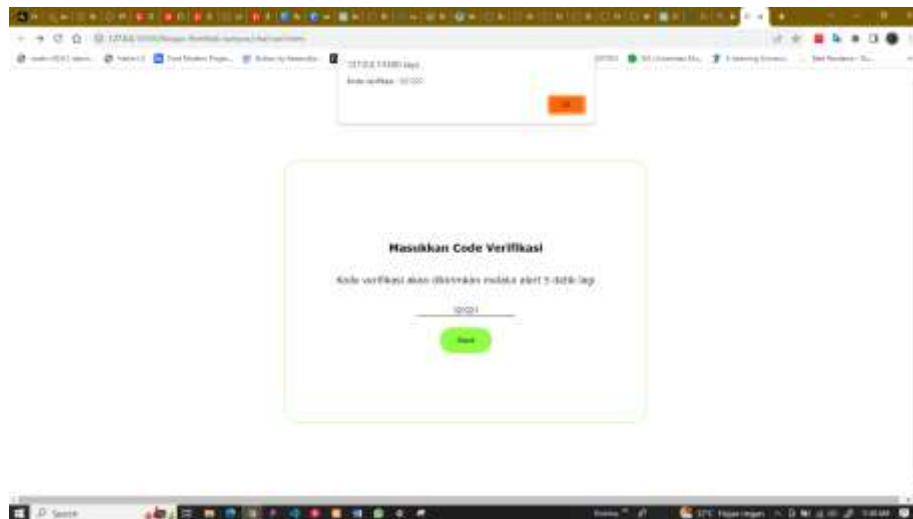
Kode CSS mendefinisikan tampilan visual aplikasi obrolan. Ini menetapkan gaya untuk berbagai elemen, termasuk:

- Gaya Umum: Gaya ini diterapkan ke semua elemen dan memastikan tata letak dan tampilan yang konsisten.
- Gaya Bagian Kiri: Gaya ini menyempurnakan daftar kontak, menjadikannya dapat diklik dan menambahkan indikasi visual untuk kontak aktif.

- Gaya Bagian Obrolan: Gaya ini menyempurnakan header, area obrolan, dan keyboard, memberikan perbedaan yang jelas antara bagian-bagian berbeda dari antarmuka obrolan.
- Gaya Pesan: Gaya ini menyempurnakan pesan obrolan individu, membuatnya terlihat berbeda dan mudah dibaca.
- Gaya Ikon: Gaya ini menerapkan pustaka ikon Font Awesome ke tombol kirim.

Membuat deisgn Verifikasi

Output:



HTML :

```
60 </head>
61 <body>
62 <div class="Form">
63 <h3>Masukkan Code Verifikasi</h3>
64 <p>Kode verifikasi akan dikirimkan melalui alert 5 detik lagi</p>
65 <input type="text" id="input" />
66 <button onclick="send()">Send</button>
67 </div>
68
```

Berisikan div class form untuk menampilkan form verifikasi

Javascript :

```
67 </div>
68
69 <script>
70   setTimeout(verif, 3000);
71   let pw = 321221;
72
73   function send() {
74     let input = document.getElementById("input");
75     if (input.value != pw) {
76       input.style.color = "red";
77       return false;
78     } else {
79       window.location.href = "message.html";
80     }
81   }
82
83   function verif() {
84     alert(`kode verifikasi : ${pw}`);
85   }
86 </script>
```

Melakukan fungsi send() yang diambil dari button, fungsinya ketika nilai inputnya tidak sama dengan code yang akan ditampilkan pada alert setelah 5 detik, maka warna font di inputan akan berubah menjadi merah.

Jika input berhasil maka akan masuk ke web message.html

CSS :

[illegible]

```

44 border-bottom: 1px solid green;
45 }
46
47 button {
48 padding: 15px 30px;
49 border-radius: 30px;
50 background-color: greenyellow;
51 border: none;
52 cursor: pointer;
53 }
54
55 button:hover {
56 background-color: black;
57 color: white;
58 }
59
60 </style>

```

Berisikan style untuk membuat form berada ditengah body dengan memiliki border radius di tiap sisinya