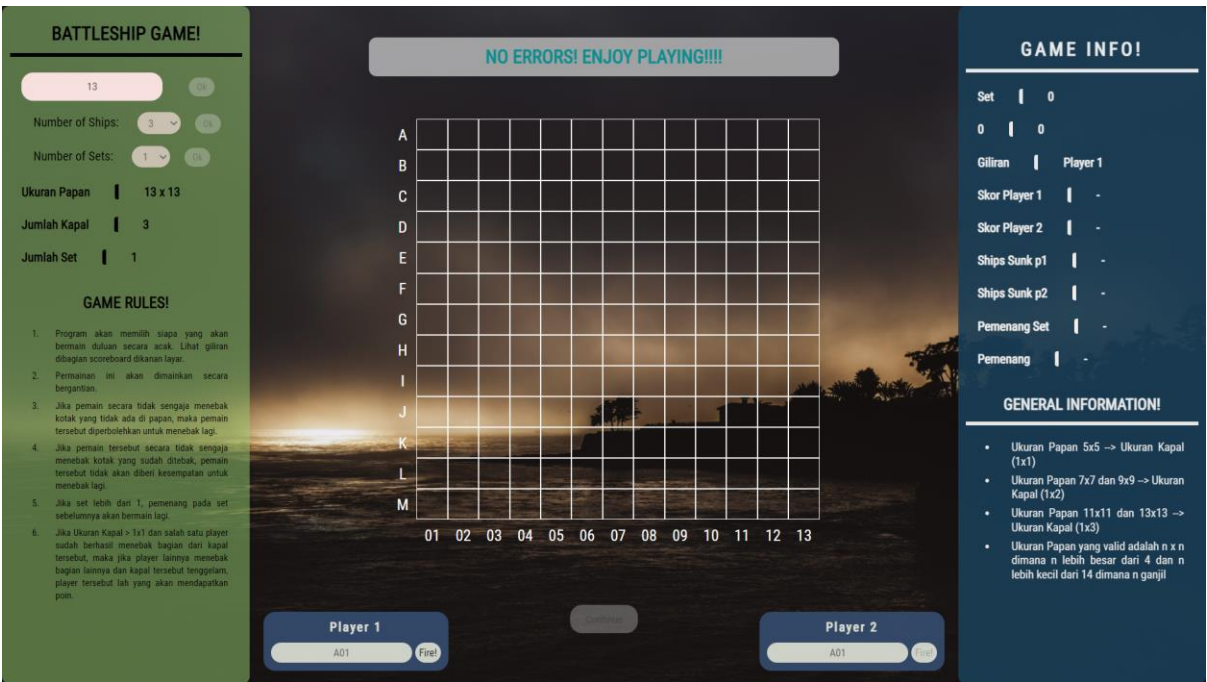
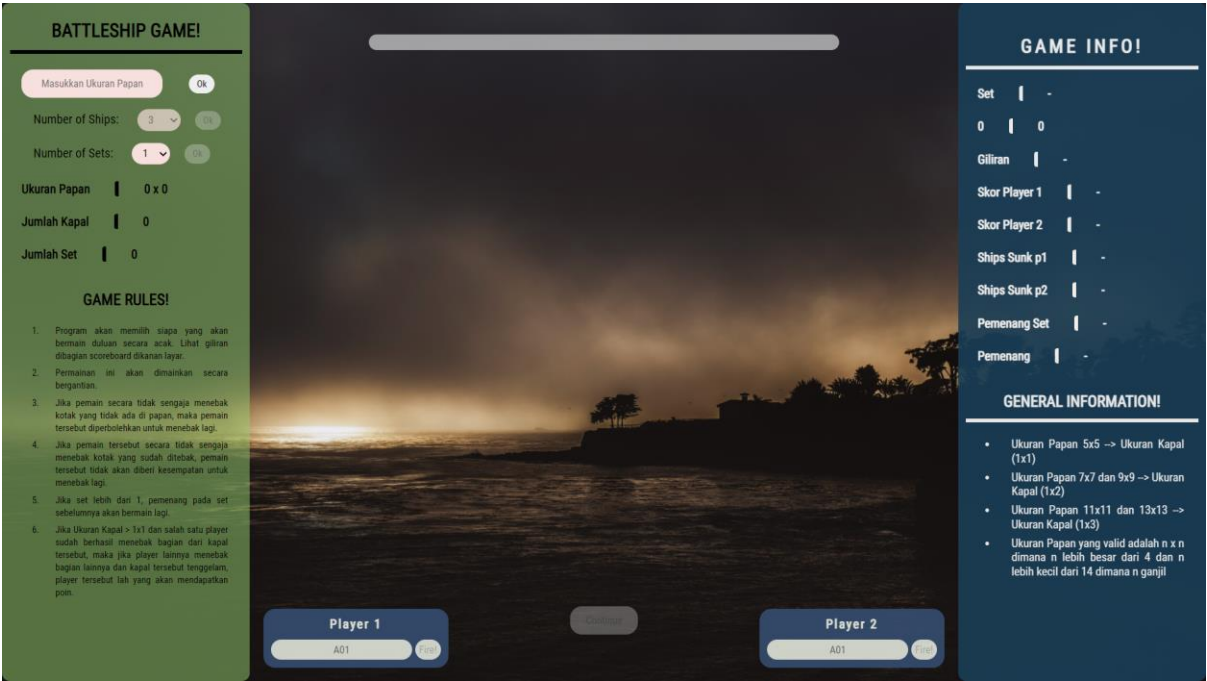


Nama: Stefannus Christian

NIM: 202000138

Prodi: IBDA

Web Layout Inti:



Penjelasan Web Layout keseluruhan:

Saya memakai display flex untuk membagi web design menjadi 3 *section* yaitu *section* kiri, *section* tengah, dan *section* kanan. *Section* kiri saya gunakan sebagai menu setting dan juga di bagian bawah *section* kiri, saya menaruh *Game Rules* yang wajib dibaca sebelum bermain permainan ini. Selain itu, bagian tengah adalah bagian dimana pemain akan memainkan *game* nya. *Section* kanan adalah *section scoreboard* dan *Game information*. Penjelasan lebih lanjut akan dijelaskan pada bagian [komponen web](#).

Komponen Web

Section leftside.

```
section#leftside {  
  background-color: rgba(152, 207, 113, 0.5);  
  padding: 0px 10px;  
  border-radius: 10px;  
}
```

- Terdapat header tag h1 Battleship Game! Yang ditaruh di paling atas section leftside. Saya menggunakan warna-warna navy atau *color palette* navy yaitu warna-warna hijau gelap, dan biru gelap. Dapat dilihat pada pengaturan background-color, saya menggunakan rgba untuk memberikan sedikit transparansi agar *background image* sedikit terlihat pada seluruh bagian dari web.

```
h1 {  
  text-align: center;  
  text-transform: uppercase;  
  border-bottom: 5px solid;  
  padding-bottom: 10px;  
}
```

- Dibawah Title Battleship Game!, terdapat form yang saya berikan input type text dan button untuk

memasukkan ukuran papan. Saya pun memberi background color dengan rgb

```
input#ukuranPapanInput {  
  margin-right: 30px;  
  margin-left: 15px;  
  background-color: rgb(248, 224, 222);  
  text-align: center;  
}  
  
#NumShipsInputButton {  
  margin-left: 15px;  
}
```

berikut karena setelah saya melakukan *searching*, warna pink cocok dengan warna hijau tua.

BATTLESHIP GAME!

Number of Ships:

Number of Sets:

Ukuran Papan

0 x 0

Jumlah Kapal

0

Jumlah Set

0

GAME RULES!

1. Program akan memilih siapa yang akan bermain duluan secara acak. Lihat giliran dibagian scoreboard di kanan layar.
2. Permainan ini akan dimainkan secara bergantian.
3. Jika pemain secara tidak sengaja menebak kotak yang tidak ada di papan, maka pemain tersebut diperbolehkan untuk menebak lagi.
4. Jika pemain tersebut secara tidak sengaja menebak kotak yang sudah ditebak, pemain tersebut tidak akan diberi kesempatan untuk menebak lagi.
5. Jika set lebih dari 1, pemenang pada set sebelumnya akan bermain lagi.
6. Jika Ukuran Kapal > 1x1 dan salah satu player sudah berhasil menebak bagian dari kapal tersebut, maka jika player lainnya menebak bagian lainnya dan kapal tersebut tenggelam, player tersebut lah yang akan mendapatkan poin.

- Selanjutnya, terdapat juga input number of ships yaitu saya menggunakan dropdown dengan tag select pada html. Saya memutuskan untuk tidak menggunakan form input lagi dengan alasan memudahkan pengecekan kesalahan input pada JavaScript.

```

get_ukuran_papan: function() {
    var ukuranPapan = document.getElementById("ukuranPapanInput");
    var ok = document.getElementById("ukuranPapanInputButton");
    this.boardSize = parseInt(ukuranPapan.value);

    if ((this.boardSize >= 5 && this.boardSize <= 13) && (this.boardSize % 2 == 1)) {
        this.make_board();
        document.getElementById("alert_message_area").innerHTML = "NO ERRORS! ENJOY PLAYING!!!"
        document.getElementById("alert_message_area").style.color = "#B5A8AA";
        ukuranPapan.disabled = true;
        ok.disabled = true;
        document.getElementById("numShips").disabled = false;
        if (this.boardSize == 5) {
            this.shipLength = 1;
        } else if (this.boardSize == 7 || this.boardSize == 9) {
            this.shipLength = 2;
        } else if (this.boardSize == 11 || this.boardSize == 13) {
            this.shipLength = 3;
        }
        document.getElementById("numShipsInputButton").disabled = false;
    } else {
        document.getElementById("alert_message_area").innerHTML = "BACA GENERAL INFORMATION!!!"
        document.getElementById("alert_message_area").style.color = "#DC412B";
        ukuranPapan.value = "";
    }
    var ukuran_papan = document.getElementById("ukuran_papan_pilihan");
    ukuran_papan.innerHTML = this.boardSize + " x " + this.boardSize;
},

```

- Gambar diatas merupakan bagaimana cara saya mengecek

apakah ukuran input papan yang di inputkan sudah benar. Pada bagian kanan dari papan akan terdapat *Game Information* yang akan memberi tahu berapa saja ukuran input papan yang valid. Dapat dilihat bahwa untuk melakukan pengecekan, dibutuhkan banyak logika if dan else yang menyebabkan kode menjadi redundant. Maka dari itu untuk menghindari hal ini, saya menggunakan select dan option saja sehingga pengecekan tidak perlu lagi dilakukan karena input dari user akan selalu sesuai dengan value yang terdapat pada select tersebut.

- Selanjutnya, saya memberikan informasi mengenai ukuran papan, jumlah kapal, dan jumlah set yang ingin dimainkan yang telah dipilih user. Hal ini saya lakukan di JavaScript dengan memanggil `document.getElementById` kemudian saya memberikan `innerHTML` sesuai dengan value input yang sudah user masukkan

BATTLESHIP GAME!

Number of Ships:

Number of Sets:

Ukuran Papan | 0 x 0

Jumlah Kapal | 0

Jumlah Set | 0

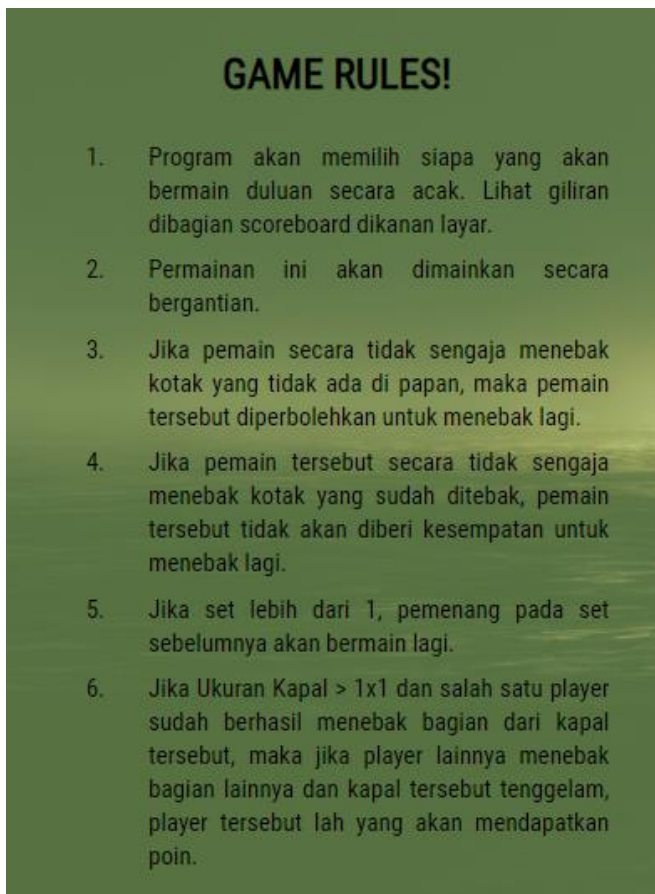
GAME RULES!

1. Program akan memilih siapa yang akan bermain duluan secara acak. Lihat giliran dibagian scoreboard dikanan layar.
2. Permainan ini akan dimainkan secara bergantian.
3. Jika pemain secara tidak sengaja menebak kotak yang tidak ada di papan, maka pemain tersebut diperbolehkan untuk menebak lagi.
4. Jika pemain tersebut secara tidak sengaja menebak kotak yang sudah ditebak, pemain tersebut tidak akan diberi kesempatan untuk menebak lagi.
5. Jika set lebih dari 1, pemenang pada set sebelumnya akan bermain lagi.
6. Jika Ukuran Kapal > 1x1 dan salah satu player sudah berhasil menebak bagian dari kapal tersebut, maka jika player lainnya menebak bagian lainnya dan kapal tersebut tenggelam, player tersebut lah yang akan mendapatkan poin.

- Selanjutnya, saya memberikan game rules untuk dibaca dulu sebelum bermain. Berikut adalah game rules nya secara lebih detail. Terdapat 6 peraturan saya buat pada game saya. Peraturan pertama adalah mengenai random turn yang saya buat pada JavaScript menggunakan Math.Random dan Math.Floor sesuai yang sudah diajarkan di buku Headfirst JavaScript dari bab 1 – 8.

```
function getRandomNumberBetween(min, max) {  
    return Math.floor(Math.random() * (max - min + 1) + min);  
}
```

Inti dari fungsi ini adalah, fungsi ini akan menggenerate random number dari 0 – 1. Jika ternyata angka yang dihasilkan adalah angka 0, maka player 1 lah yang akan jalan terlebih dahulu, sebaliknya jika angka yang dihasilkan adalah angka 1, maka player 2 lah yang akan bermain terlebih dahulu.



Tambahan lagi, setelah player memilih ukuran papan, maka program JavaScript akan auto menggenerate papan permainan yang masing-masing diberi ID A01,A02,.....A0N, B01,B02,.....B0N.....N0N dimana N adalah ukuran papan yang valid.


```

make_board: function() {
    var papan = document.getElementById("pa pan");
    var alphabet = ["A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M"];
    var board = [];
    for (var i = 0; i < this.boardSize; i++) {
        var row = papan.insertRow(i);
        for (var j = 0; j < this.boardSize; j++) {
            var col = row.insertCell(j);
            if (j < 9) {
                col.setAttribute("id", alphabet[i] + "" + "0" + (j + 1));
                board.push(alphabet[i] + "" + "0" + (j + 1));
            } else {
                col.setAttribute("id", alphabet[i] + (j + 1));
                board.push(alphabet[i] + (j + 1));
            }
        }
        var col = row.insertCell(0);
        col.setAttribute("class", "axis");
        col.innerHTML = alphabet[i];
    }
}

```

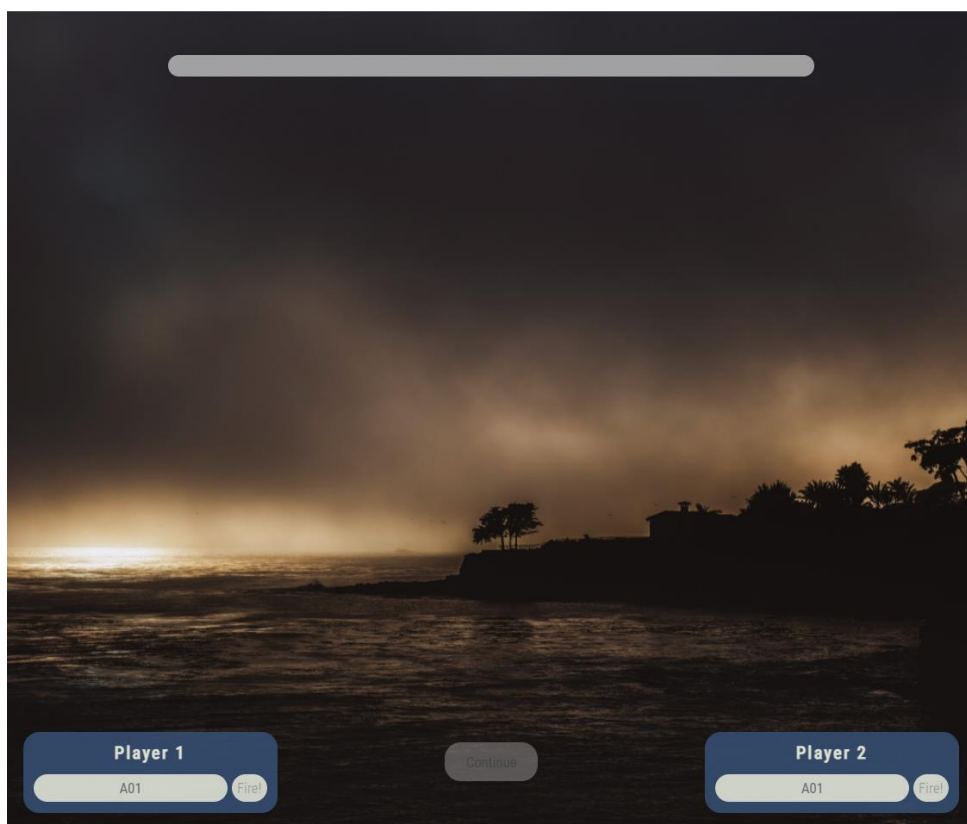
Ini merupakan algoritma untuk menggenerate papan nya. Kemudian saya menambahkan satu kolom dan satu baris tanpa border yang digunakan untuk menamai masing-masing papan sehingga player tidak kebingungan ketika menebak.

Section

Board

/

Tengah.

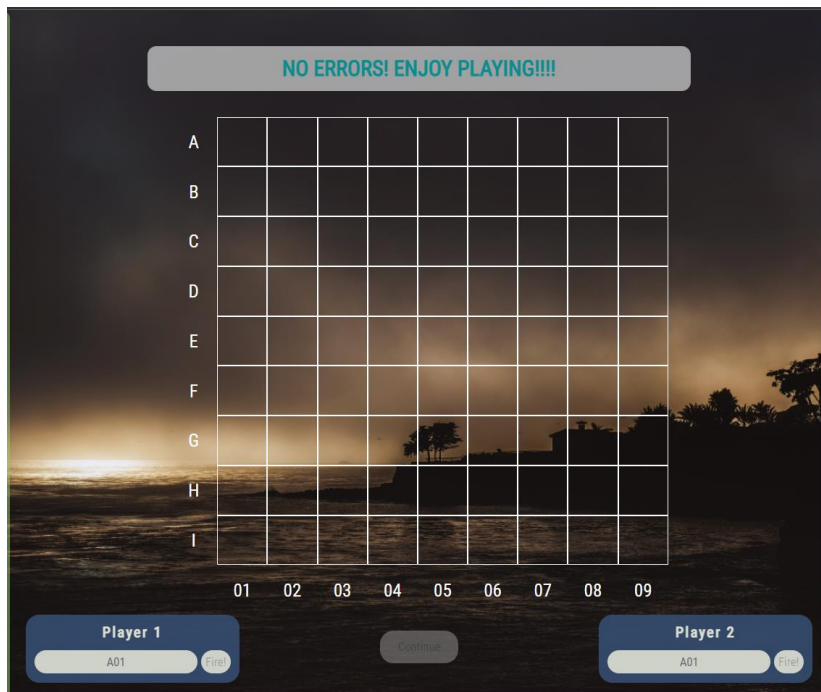


Ini merupakan tampilan paling sederhana dari bagian tengah papan ketika player belum memasukkan input apapun. Di bagian atas terdapat suatu area berwarna abu-abu yang saya

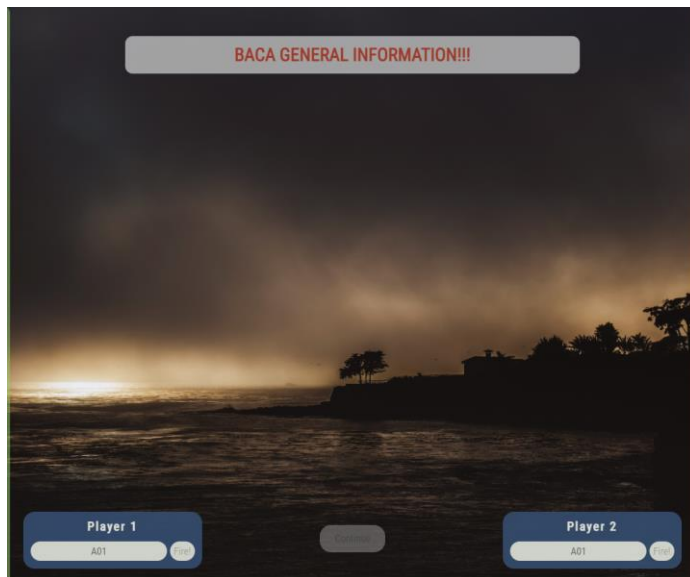
gunakan untuk display message ketika ada kapal yang tertembak, tenggelam, invalid input, dll. Di bagian pojok kiri saya gunakan untuk input player 1 dan di bagian kanan bawah untuk player 2 input.

```
h1#alert_message_area {  
  display: inline-block;  
  border-bottom: none;  
  background-color: #C1C1C1;  
  opacity: 0.8;  
  width: 600px;  
  border-radius: 10px;  
  padding-top: 10px;  
}
```

Ini merupakan CSS yang saya gunakan untuk menaruh display message nya.

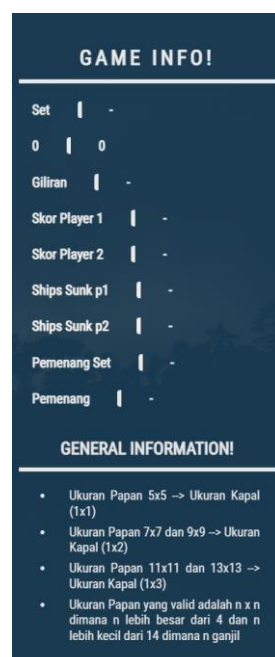


Ini merupakan tampilan papan ketika player memasukkan 9x9 sebagai ukuran papan.

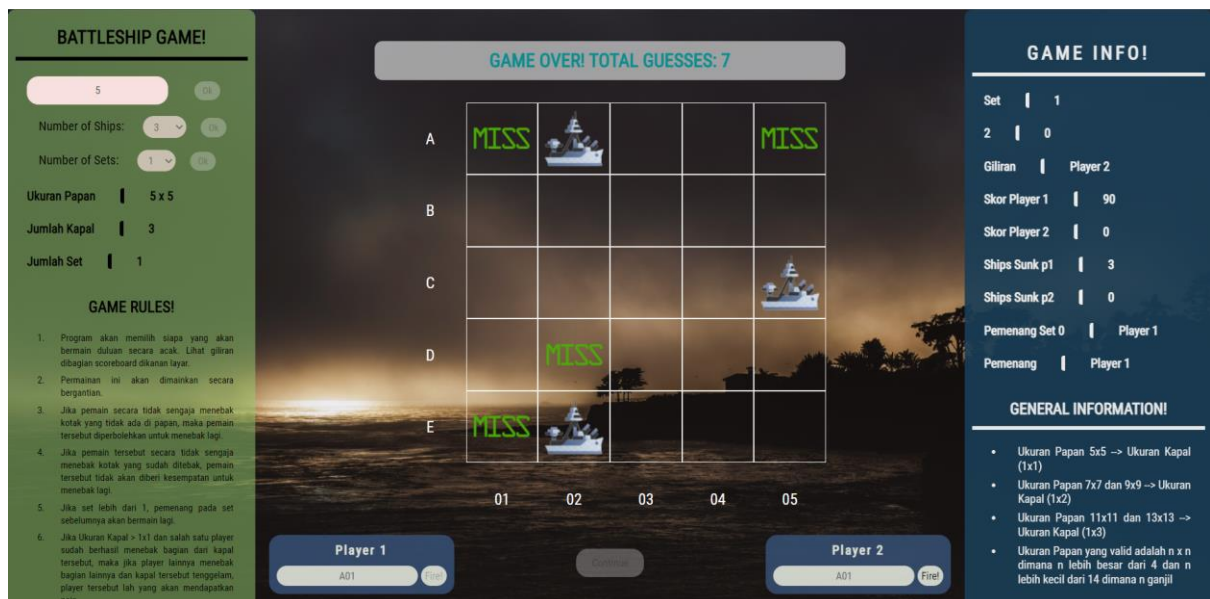
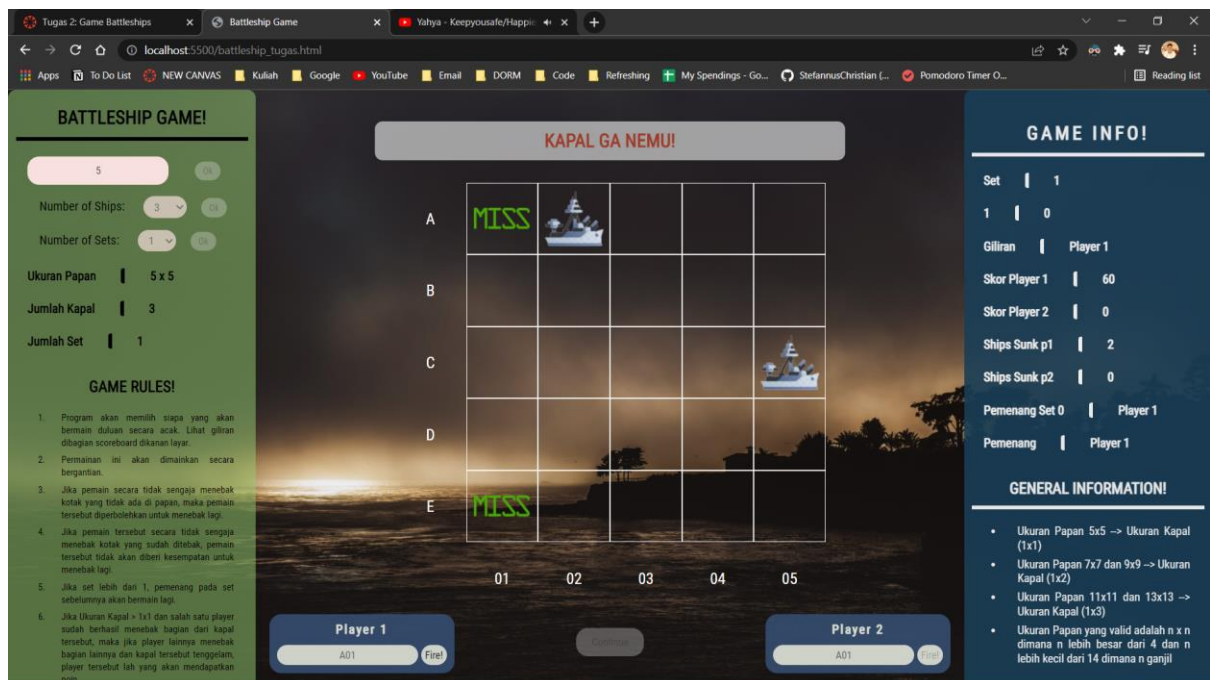


Ini merupakan tampilan saat player salah memasukkan ukuran papan. Disitu tertera tulisan baca general information karena pada bagian general information yang saya letakkan di bagian kanan, terdapat ukuran papan input yang valid.

Pada bagian section kanan ini saya menaruh scoreboard dan general information yang wajib dibaca sebelum bermain agar player tidak kebingungan. Semua bagian dari GAME INFO! Akan ter update otomatis ketika game sedang berjalan. Semuanya ini dilakukan dengan logika JavaScript dimana saya get element by id dan saya mengambil value nya kemudian saya mengupdate inner html dari masing -masing header yang saya sudah berikan ID di HTML



DEMO PERMAINAN.



Berikut merupakan demo permainan dimana set yang dipilih adalah 3. Setelah set selesai dapat dilihat bahwa terdapat informasi mengenai siapa yang menang di set tersebut dan dapat dilihat bahwa scoreboard di kanan ter update. Terdapat satu lagi fitur yaitu tombol continue. Tombol continue ini awalnya saya disable dan akan saya enabled jika Set nya lebih dari satu.