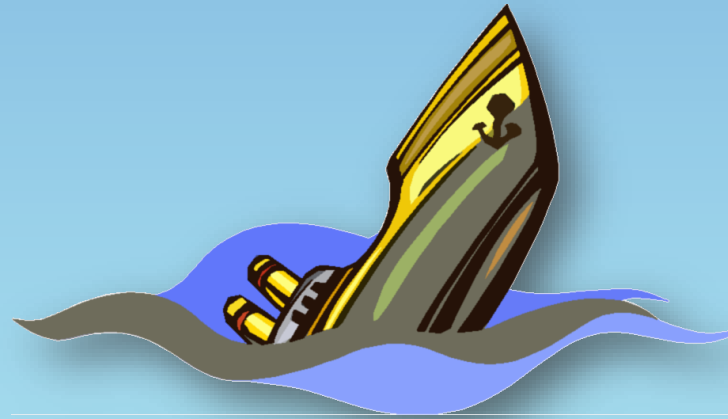


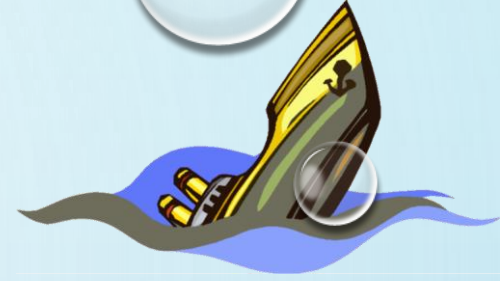
# ENTWICKLUNG EINES 2D – „SCHIFFE VERSENKEN“ SPIELS ZUR SIMULATION EINER SEESCHLACHT



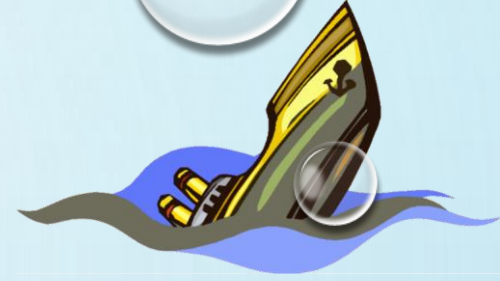
DENIS OJDANIC

# INHALT

- VORSTELLUNG
- AUFTRAGGEBER
- PROJEKTZIEL
- PLANUNG
- DURCHFÜHRUNG
- QUALITÄTSSICHERUNG
- FAZIT
- DEMONSTRATIONSVIDEO
- QUELLEN



# VORSTELLUNG



NAME: DENIS OJDANIC

HÖCHSTER ABSCHLUSS: FACHHOCHSCHULREIFE BEI PRO GENIUS IN HEIDENHEIM

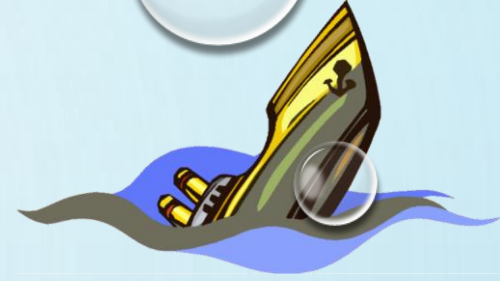
GEBOREN: 04.10.1994 IN HEIDENHEIM AN DER BRENZ

AUSBILDUNG: FACHINFORMATIKER – ANWENDUNGSENTWICKLUNG

FERTIGKEITEN: WEBENTWICKLUNG MIT HTML, CSS, JAVASCRIPT & PHP

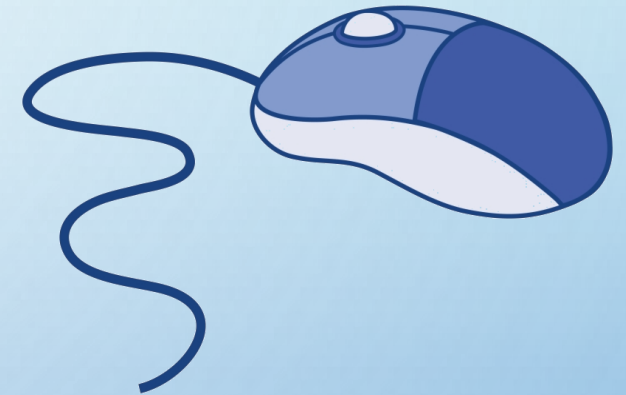
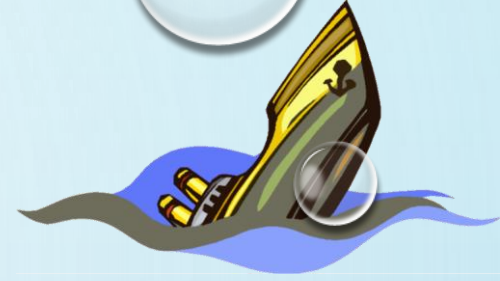
# AUFTRAG GEBER

- OJDE-SCHULE IN HEIDENHEIM (BRENZ)
- FÖRDERSCHULE MIT FÖRDERSCHWERPUNKT KÖRPERLICHE UND MOTORISCHE ENTWICKLUNG
- GRUND- UND HAUPTSCHULE
- 200 SCHÜLER MIT KLEINEN KLASSENGRÖßEN
- PROJEKTEINSATZ: SCHULUNTERRICHT FACH „EDV“



# PROJEKTZIEL

- ENTWICKLUNG EINES „SCHIFFE VERSENKEN“-SPIELS
- SCHÜLER-PCS DURCH ZUGRIFF AUF WEBSERVER DER OJDE-SCHULE
- BROWSER MOZILLA FIREFOX UND GOOGLE CHROME
- DEMOVERSION FÜR NUR EIN SPIELVORGANG
- BEDIENUNG DES SPIELS MIT COMPUTER MAUS
- MOTORISCH EINGESCHRÄNKTE SCHÜLER ALS BENUTZER  
→ FÄHIGKEITSVERBESSERUNG AUF LANGE SICHT





# PROJEKTZIEL

## Schiffe versenken

10 Schiffe zum Start:

### FEATURES

x	x	x	x	x
---	---	---	---	---

1x Schlachtschiff  
5 Kästchen

x	x	x	x
---	---	---	---

2x Kreuzer  
4 Kästchen

x	x	x
---	---	---

3x Zerstörer  
3 Kästchen

x	x
---	---

4x U-Boot  
2 Kästchen

### Eigene Schiffe:

	1	2	3	4	5	6	7	8	9	10	
A											A
B											B
C											C
D											D
E											E
F											F
G											G
H											H
I											I
J											J
	1	2	3	4	5	6	7	8	9	10	

- SPIELER GEGEN SPIELER
- ONLINE ÜBER EINEN WEBSERVER NUTZBAR
- STATISTIK FÜR ZÜGE, TREFFERQUOTEN UND VERLORENE SCHIFFE

### REGELN

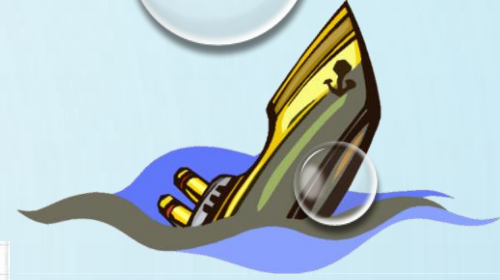
### Gegnerische Schiffe:

1. SPIELFELDGRÖßE: 10 x 10 FELDER

	1	2	3	4	5	6	7	8	9	10	
A											A
B											B
C											C
D											D
E											E
F											F
G											G
H											H
I											I
J											J
	1	2	3	4	5	6	7	8	9	10	

2. ZUFÄLLIGE AUSWAHL FÜR SPIELSTART
3. KEIN ANENANDERSTOßEN DER SCHIFFE
4. SCHIFFE NICHT ÜBER ECK GEBAUT, KEINE AUSBUCHTUNGEN
5. SCHIFFPLATZIERUNG AM RAND DES SPIELFELDS MÖGLICH
6. KEINE DIAGONALE SCHIFFSPLATZIERUNG
7. ZEHN SCHIFFE PRO SPIELER

# PLANUNG

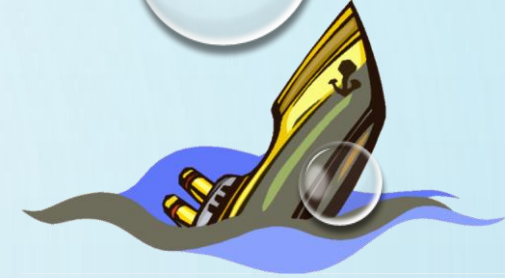


Projektphasen		geplante Zeit
Sp	Kundengespräch <i>VS</i>	<i>Spieler 2</i> 1 h
	Projektplanung	4 h
	GUI-Entwurf	4 h
	Erstellung der GUI <i>Schiffe:</i>	6 h
	Implementierung der Features <i>x/1</i>	12 h
	Implementierung der Regeln <i>x/2</i>	16 h
	Funktionsprüfung & Qualitätskontrolle <i>x/3</i>	4 h
	Soll-Ist-Vergleich <i>x/4</i>	1 h
	Abnahme	2 h
	Dokumentation	30 h
	Summe	80 h

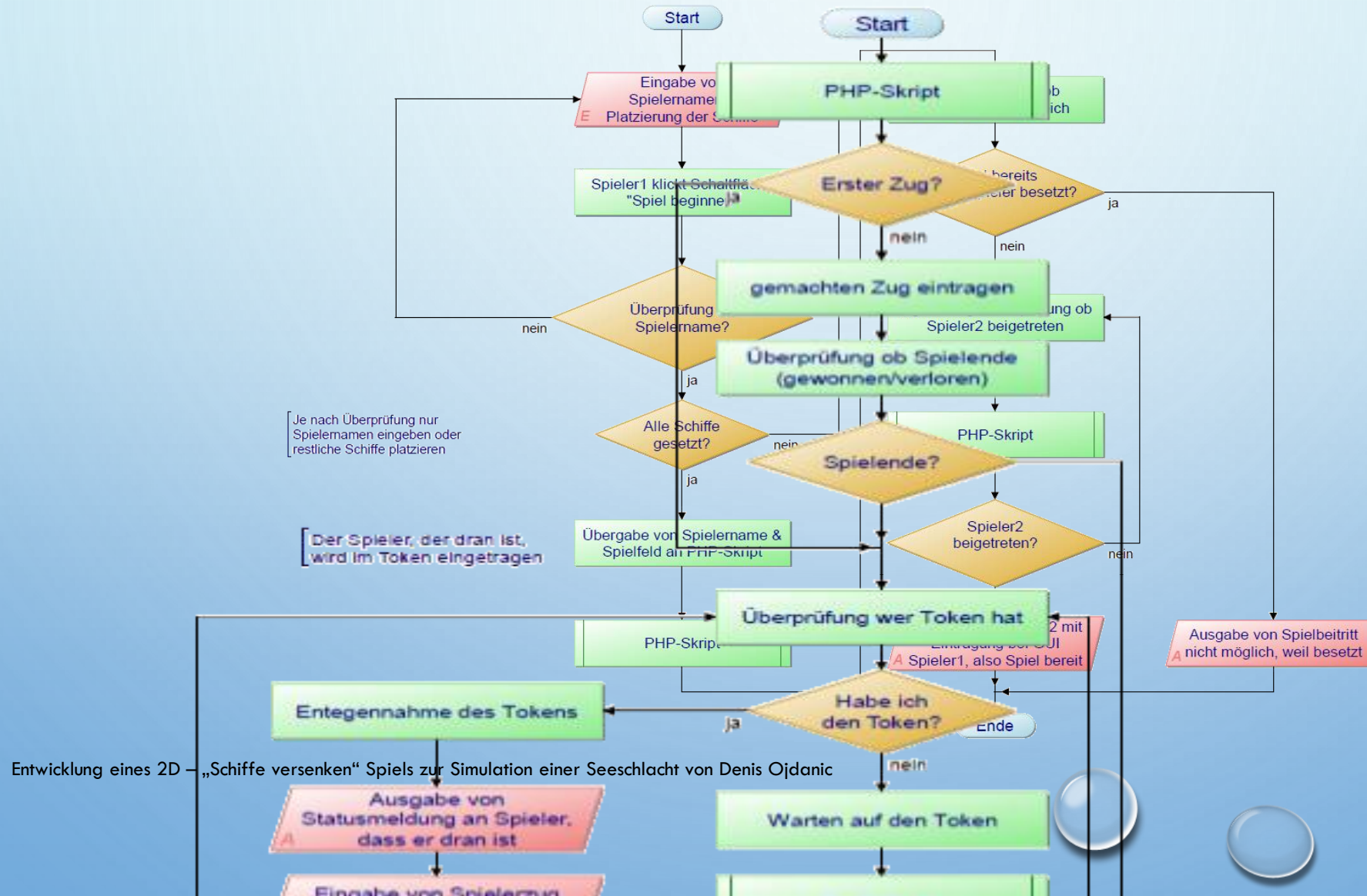
*Spiel beginnen*



# PLANUNG

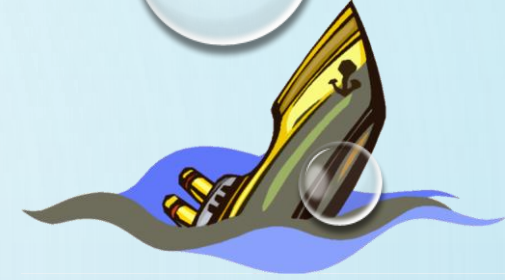


Registrierungsvorgang zum Spielvorgang





# DURCHFÜHRUNG: OBERFLÄCHE



## 1. ERSTELLUNG DER WEBOBERFLÄCHE

vs

Schiffe platzieren.

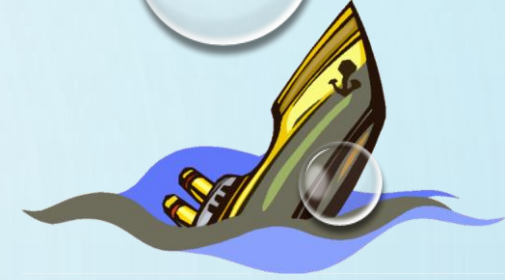
	1	2	3	4	5	6	7	8	9	10	
A	W	W	W	W	W	W	W	W	X	○	A
B	W	X	X	X	X	X	W	W	X	○	B
C	W	X	○	○	○	X	W	W	X	X	C
D	W	X	X	X	X	X	W	W	W	W	D
E	W	W	W	W	W	W	W	W	W	■	E
F	W	W	W	W	W	W	W	W	W	W	F
G	W	W	W	W	W	W	W	W	W	W	G
H	X	X	X	X	X	X	X	W	W	W	H
I	X	○	○	○	○	○	X	W	W	W	I
J	X	X	X	X	X	X	X	W	W	W	J
	1	2	3	4	5	6	7	8	9	10	

**Schiffe:**  
■■■■■ 0 / 1  
■■■■ 0 / 2  
■■■ 0 / 3  
■■ 0 / 4

	1	2	3	4	5	6	7	8	9	10	
A	?	?	?	?	?	?	?	?	?	?	A
B	?	?	?	?	?	?	?	?	?	?	B
C	?	?	?	?	?	?	?	?	?	?	C
D	?	?	?	?	?	?	?	?	?	?	D
E	?	?	?	?	?	?	?	?	?	?	E
F	?	?	?	?	?	?	?	?	?	?	F
G	?	?	?	?	?	?	?	?	?	?	G
H	?	?	?	?	?	?	?	?	?	?	H
I	?	?	?	?	?	?	?	?	?	?	I
J	?	?	?	?	?	?	?	?	?	?	J
	1	2	3	4	5	6	7	8	9	10	

Spiel beginnen

# DURCHFÜHRUNG: TEXTDATEI

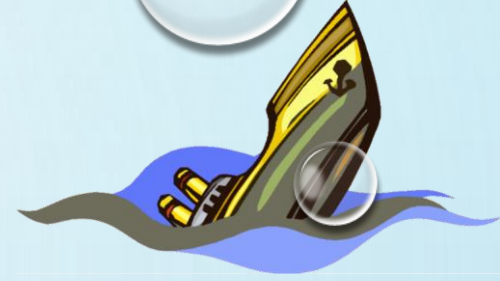


## 2. AUFBAU DER TEXTDATEI ALS JSON-STRING

- SPIELERNAMEN
- TOKEN
  - SPIELER MIT ZUG
  - LETZTER SPIELZUG (ZUSATZ)
- SCHIFFAUFSTELLUNGEN

```
{  
  "spieler1": "Rudolf",  
  "spieler2": "Bambi",  
  "token": "Spieler1;",  
  "aufstellung1": [  
    ["a1", "b1"],  
    ["f1", "g1", "h1", "i1", "j1"],  
    ["d1", "d2", "d3", "d4"],  
    ["d6", "d7", "d8"],  
    ["c10", "d10", "e10"],  
    ["a4", "a5", "a6"],  
    ["g4", "g5"],  
    ["f7", "g7", "h7", "i7"],  
    ["j9", "j10"],  
    ["i4", "j4"]  
  ],  
  "aufstellung2": [  
    ["a10", "b10", "c10", "d10", "e10"],  
    ["g10", "h10", "i10", "j10"],  
    ["i1", "i2", "i3", "i4"],  
    ["a1", "a2", "a3"],  
    ["f5", "f6", "f7"],  
    ["h7", "i7"],  
    ["a7", "b7"],  
    ["d2", "d3", "d4"],  
    ["f2", "g2"],  
    ["d7", "d8"]  
  ]  
}
```

# DURCHFÜHRUNG: PHASEN

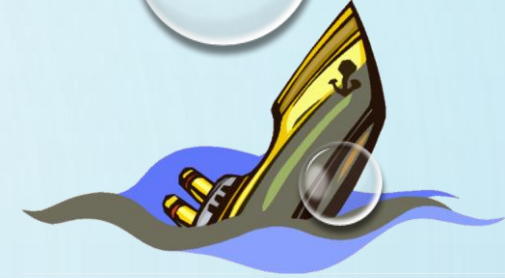


## 3. PHASEN DER IMPLEMENTIERUNG DER SPIELLOGIK

- 1) PLATZIERUNG DER SCHIFFE
- 2) ANMELDUNG DER SPIELER
- 3) ABLAUF DER SPIELZÜGE
- 4) AUSGABE DER STATISTIK



# DURCHFÜHRUNG: PLATZIERUNG DER SCHIFFE

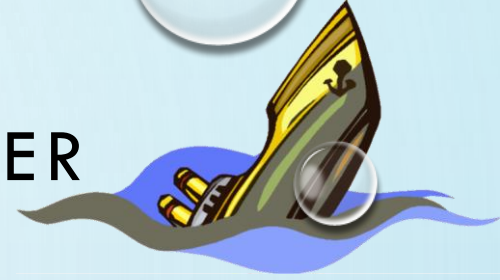


```
// benötigte Elemente in schiffsfläche laden + sperrbereich neben Schiff befüllen
for (let zeichne = 0; wert > zeichne; zeichne++)
{
  let IDsperrDavor = "";
  let IDsperrDanach = "";
  //console.log("Start: " + start); // Testausgabe
  let IDzusatz = "";
  if (ausrichtung === "h") // horizontal
  {
    let temp = zeichne + parseInt(start.slice(1)); // zur vorbereitung der variablen Zahl der ID
    //console.log("Temp: " + temp); // Testausgabe
    IDzusatz = start.charAt(0) + temp; // für die Fläche des Schiffs
    //console.log("IDzusatz: " + IDzusatz);
    IDsperrDavor = String.fromCharCode(start.charCodeAt(0) - 1) + temp;
    IDsperrDanach = String.fromCharCode(start.charCodeAt(0) + 1) + temp;
    //console.log("von " + IDsperrDavor + " bis " + IDsperrDanach); // Testausgabe für davor und danach sperren
  }
  else // vertikal
  {
    let temp = String.fromCharCode(zeichne + start.charCodeAt(0)); // zur vorbereitung des variablen Buchstabens der ID
    //console.log("Temp: " + temp); // Testausgabe
    IDzusatz = temp + start.slice(1); // für die Fläche des Schiffs
    //console.log("IDzusatz: " + IDzusatz);
    IDsperrDavor = temp + (start.slice(1) - 1);
    IDsperrDanach = temp + (parseInt(start.slice(1)) + 1);
    //console.log("von " + IDsperrDavor + " bis " + IDsperrDanach); // Testausgabe für davor und danach sperren
  }
  //console.log("schiffsfläche..."); // Testausgabe
}
```

- AUSSCHNITT AUS FUNKTION PLATZIERESCHIFF
- GENERIERUNG DER BENÖTIGTEN IDS FÜR SCHIFFSFLÄCHEN & SPERRFLÄCHEN



# DURCHFÜHRUNG: ANMELDUNG DER SPIELER



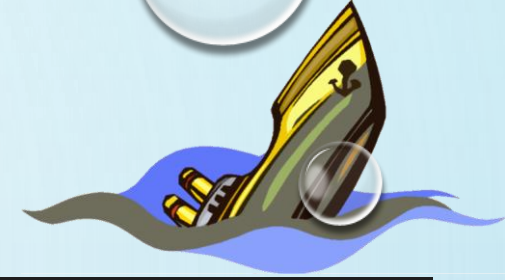
- AUSSCHNITT AUS FUNKTION REGISTRIERUNG
- ERFOLGREICHE REGISTRIERUNG VON SPIELER 2

```
else
{
    $spiel->spieler2 = $_POST["spieler"];
    $json = json_decode($_POST["aufstellung"]); # String in Array umwandeln
    $spiel->aufstellung2 = $json; # Array eintragen
    #print_r($spiel); # Testausgabe
    echo "Spieler2;registriert.;Gegner;"; # Ausgabe für JS, getrennt durch ";"
    echo $spiel->spieler1 . ";"; # Spieler 1 Namen zurückgeben
    #$spiel->token = "test123"; # Testeintrag in JSON

    if (rand(0, 1) === 0) # Zufallsentscheid, wer anfängt von 0 bis 1 (Ganzzahl)
    {
        $spiel->token = "Spieler1"; # ID 1. Spieler
    }
    else
    {
        $spiel->token = "Spieler2"; # ID 2. Spieler
    }
    $spiel->token .= ";"; # danach kommt letzter gemachter Zug, bei Erstzug leer

    echo $spiel->token; # Rückgabe des Tokens
    $json = json_encode($spiel); # JSON String generieren
    file_put_contents($datei, $json); # in Datei schreiben
    #echo rand(0,1); # Spieler bestimmen der anfängt
}
```

# DURCHFÜHRUNG: SPIELZÜGE

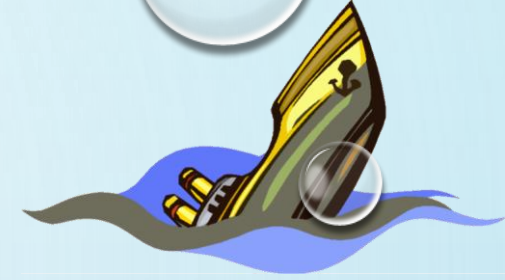


- AUSSCHNITT AUS FUNKTION AUSWERTEN
- ÜBERPRÜFUNG DER AUFSTELLUNG BEI SPIELZUG NACH TREFFER

```
foreach ($schiffsammlung as &$schiff) # & => Referenz
{
    foreach ($schiff as &$feld) # & => Referenz
    {
        # prüfen ob getroffen; übergebe Array-Index -> $treffer
        if (($treffer = array_search($spielzug, $schiff)) !== false)
        {
            #print_r($schiff); # Testausgabe
            #echo "Treffer: ".$treffer; # Testazsgabe
            unset($schiff[$treffer]); # Feld aus Schiff-Array nehmen
            $schiff = array_values($schiff); # Entfernen von Index (alle)
            $status = "T"; # (T)reffer

            if (count($schiff) === 0) # prüfen ob keine mehr beschießbaren Felder
            {
                $status = "V"; # (V)ersenkt
            }
            #print_r($schiff); # Testausgabe
        }
    }
    if ($leer) # solange leer ist überprüfe, ob Schiff leer
    {
        $leer = empty($schiff); # nicht leeres Schiff gefunden
    }
    #print_r($schiffsammlung); # Testausgabe
}
```

# DURCHFÜHRUNG: STATISTIK



## BERECHNUNG UND EINTRAG DER STATISTIK UNTER DER SCHIFFANZEIGE

Outlook   Schiffe versenken

Alle gegnerischen Schiffe wurden versenkt!

	1	2	3	4	5	6	7	8	9	10	
A	X	X	X	W	X	W	W	W	X	X	A

Schiff

Auf localhost wird Folgendes angezeigt:  
Spiel gewonnen!

Ok

	1	2	3	4	5	6	7	8	9	10	
A	T	W	W	V	W	W	W	V	T	T	A

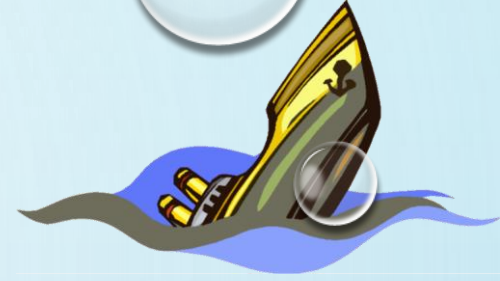
```
function statistikausgabe() // Eintrag in Statistikbereich unter der Schiffanzeige
{
    let ausgabe = "<br><br>" + treffer + " Treffer von " + zuganzahl + " Schüssen<br>"; // x Treffer von y Schüssen
    ausgabe += "<br>Eigene Trefferquote: " + parseInt(100 * treffer / zuganzahl) + " %"; // ganze % ohne Kommastellen

    document.getElementById("statistik").innerHTML = ausgabe;
}
```

H	X	W	X	W	W	W	W	X	W	X	H		H	T	W	W	W	W	W	W	W	W	H	
I	X	W	X	X	X	W	W	W	W	X	I		I	T	W	W	T	T	V	W	W	W	I	
J	X	X	X	X	X	W	X	X	X	X	J		J	T	W	W	W	W	W	W	W	V	T	J
	1	2	3	4	5	6	7	8	9	10					1	2	3	4	5	6	7	8	9	10

Spiel beginnen

# QUALITÄTSSICHERUNG



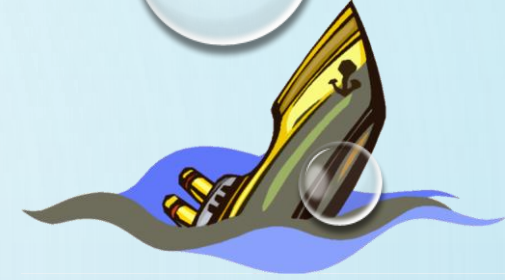
- FOKUS DER DEMOVERSION → FUNKTIONSFÄHIGKEIT
- UNITTESTS NACH JEDER ERSTELLTEN FUNKTION
- WHITE-BOX-TEST NACH FERTIGSTELLUNG
- PRAKTISCHER BLACKBOX-TEST
- ENDBENUTZERTEST STEHT NOCH AUS

## QUALITÄTSSICHERUNG





# FAZIT



## GELERNTES:

- MEHR ÜBER WEBENTWICKLUNG GELERNT
- FEHLER RECHT SCHNELL BEHOBEN
- PLANUNG IN ZUKUNFT PRÄZISER

## AUSBLICK:

- ANPASSUNGEN IM DESIGN
- AUSWERTUNGEN ÜBERWIEGEND DURCH SERVER
- SIMULTAN STATTFINDENDE SPIELMATCHES

Projektphasen	geplante Zeit	benötigte Zeit	Differenz
Kundengespräch	1 h	1 h	0 h
Projektplanung	4 h	4 h	0 h
GUI-Entwurf	4 h	4 h	0 h
Erstellung der GUI	6 h	5 h	-1 h
Implementierung der Features	12 h	20 h	+8 h
Implementierung der Regeln	16 h	9 h	-7 h
Funktionsprüfung & Qualitätskontrolle	4 h	4 h	0 h
Soll-Ist-Vergleich	1 h	1 h	0 h
Abnahme	2 h	2 h	0 h
Dokumentation	30 h	30 h	0 h
Summe	80 h	80 h	0 h

Schiffe versenken

localhost/Webseiten/Abschlussprojekt/Schiffe-versenken/Dateien/

Spielername

vs

Schiffe platzieren.

	1	2	3	4	5	6	7	8	9	10	
A	W	W	W	W	W	W	W	W	W	W	A
B	W	W	W	W	W	W	W	W	W	W	B
C	W	W	W	W	W	W	W	W	W	W	C
D	W	W	W	W	W	W	W	W	W	W	D
E	W	W	W	W	W	W	W	W	W	W	E
F	W	W	W	W	W	W	W	W	W	W	F
G	W	W	W	W	W	W	W	W	W	W	G
H	W	W	W	W	W	W	W	W	W	W	H
I	W	W	W	W	W	W	W	W	W	W	I
J	W	W	W	W	W	W	W	W	W	W	J
	1	2	3	4	5	6	7	8	9	10	

Schiffe:

0 / 1

0 / 2

0 / 3

0 / 4

Spiel beginnen

Schiffe versenken

localhost/Webseiten/Abschlussprojekt/Schiffe-versenken/Dateien/

Spielername

vs

Schiffe platzieren.

	1	2	3	4	5	6	7	8	9	10	
A	W	W	W	W	W	W	W	W	W	W	A
B	W	W	W	W	W	W	W	W	W	W	B
C	W	W	W	W	W	W	W	W	W	W	C
D	W	W	W	W	W	W	W	W	W	W	D
E	W	W	W	W	W	W	W	W	W	W	E
F	W	W	W	W	W	W	W	W	W	W	F
G	W	W	W	W	W	W	W	W	W	W	G
H	W	W	W	W	W	W	W	W	W	W	H
I	W	W	W	W	W	W	W	W	W	W	I
J	W	W	W	W	W	W	W	W	W	W	J
	1	2	3	4	5	6	7	8	9	10	

Schiffe:

0 / 1

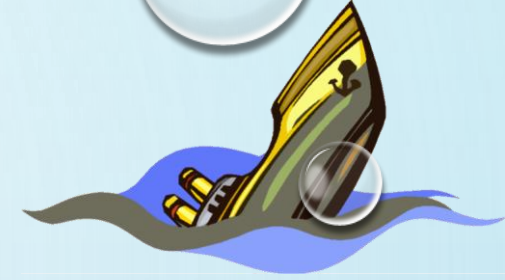
0 / 2

0 / 3

0 / 4

Spiel beginnen

# QUELLEN



- PROGRAMMABLAUFPLÄNE ERSTELLT MIT PAP DESIGNER
- DEMONSTRATIONSVIDEO ERSTELLT MIT OPEN BROADCASTER SOFTWARE®
- [HTTPS://WWW.ZITRONENBANDE.DE/WP-CONTENT/UPLOADS/2022/03/SCHIFFE-VERSENKEN-VORLAGE-1024x722.JPG](https://www.zitronenbande.de/wp-content/uploads/2022/03/schiffe-versenken-vorlage-1024x722.jpg)
- [HTTPS://WWW.XPRON.COM/WP-CONTENT/UPLOADS/2022/12/QUALITAETSSICHERUNG.JPG](https://www.xpron.com/wp-content/uploads/2022/12/Qualitaetssicherung.jpg)
- [HTTPS://ENCRYPTED-TBN0.GSTATIC.COM/IMAGES?Q=TBN:ANd9GcTtZKHdRkYBH-SGT-PODWQHANYV2QQ2RkUp9Y8BD9C3ZWAlQ2X\\_V6UZWUDWDKVF-RH88CI&USQP=CAU](https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTtZKHdRkYBH-SGT-PODWQHANYV2QQ2RkUp9Y8BD9C3ZWAlQ2X_V6UZWUDWDKVF-RH88CI&USQP=CAU)
- [HTTPS://WWW.NICEPNG.COM/PNG/DETAIL/11-111777\\_BLUE-COMPUTER-MOUSE-COMPUTER-MOUSE-CLIPART.PNG](https://www.nicepng.com/png/detail/11-111777_blue-computer-mouse-computer-mouse-clipart.png)
- [HTTPS://ENCRYPTED-TBN2.GSTATIC.COM/IMAGES?Q=TBN:ANd9GcSFRPC-Ko3GYG1zsErnJcDDEFCUQ43L90sZQsKIU4LY6-QOZ4KX](https://encrypted-tbn2.gstatic.com/images?q=tbn:ANd9GcSFRPC-Ko3GYG1zsErnJcDDEFCUQ43L90sZQsKIU4LY6-QOZ4KX)
- [HTTPS://CDN3.VECTORSTOCK.COM/I/1000x1000/16/77/LAPTOP-PROGRAMMING-CODE-WEB-TECHNOLOGY-VECTOR-21981677.JPG](https://cdn3.vectorstock.com/i/1000x1000/16/77/laptop-programming-code-web-technology-vector-21981677.jpg)