

# SaraZarubica\_LBM151\_V232

## Inhalt

Allgemein.....	2
Sicherheitsaspekte .....	2
Aufgabe 1 .....	4
Aufgabe 2 .....	4
Aufgabe 3 .....	4
Aufgabe 4 .....	5
Aufgabe 5 .....	6
Prototyp Quiz-Seite Kategorie wählen .....	6
Prototyp Quiz-Seite .....	6
Prototyp Quiz-Seite Name für Highscoreliste .....	7
Prototyp Quiz-Seite Highscoreliste.....	7
Prototyp Admininterface Startseite .....	8
Prototyp Admininterface Fragenliste.....	8
Prototyp Admininterface Fragenbearbeitung.....	9
Prototyp Admininterface Kategorienliste .....	10
Prototyp Admininterface Kategorienbearbeitung .....	10
Prototyp Admininterface Highscore löschen .....	11
Aufgabe 6 .....	12
Aufgabe 7 .....	13
Abnahmeprotokoll .....	13
Aufgabe 7 .....	<b>Error! Bookmark not defined.</b>
Aufgabe 8 .....	16
Testumgebung.....	16
Testfallspezifikation.....	16
Testfall Protokoll .....	20
Fazit der Tests.....	20
Aufgabe 9 .....	20

## Allgemein

Das Projekt wurde im Rahmen der LB 151 durchgeführt. Und in Google Chrome getestet.

Link: <https://wmm.messerli.ch>

Wichtig (Nur über https erreichbar)

Benutzerdaten:

Username: admin

Passwort: admin1234

## Sicherheitsaspekte

### *Transaktionen*

Transaktionen werden im Entity Framework per Standard gemacht. Wenn SaveChanges() zum Einfügen, Aktualisieren oder Löschen auf der Datenbank ausgeführt wird, fügt das Framework diesen Vorgang in eine Transaktion ein.

### *Passwortabfrage*

Das Passwort wird durch Salts geschützt. Unter Salt versteht man das Hinzufügen von weiteren Zeichenfunktionen in den Passworthash. Gehashed wird das Passwort mit dem Sha256 Algorithmus. Salt benutzt man, da es Angriffe, wie zum Beispiel den Wörterbuchangriff durch Rainbow Tables erschwert.

Siehe Beispiel nächste Seite.

## Registrierung

Wenn sich ein Benutzer registrieren will wird ein Salt generiert. Danach wandelt man das Passwort des Benutzers zu einem Bytearray um. Das umgewandelte Passwort, wird dann mit dem Salt gehashed und am Schluss gespeichert.

Siehe Code Beispiel:

```
1 reference | SaraZarubica, 35 days ago | 1 author, 1 change | 0 exceptions
public bool Register(string userName, string password, Roll roll)
{
    byte[] salt = PasswordHelper.GetSalt(128); // salt generieren
    byte[] passwordSalted = PasswordHelper.GenerateSaltedHash(StringUtils.GetByteArray(password), salt); //pw to bytearray, pw byte array mit salt hashen
    User user = new User() { Password = passwordSalted, Salt = salt, Username = userName, Roll = roll }; // salted pw wird gespeichert
    if(dbContext.Users.Where(x => x.Username.ToLower() == userName.ToLower()).FirstOrDefault() == null)
    {
        dbContext.Users.Add(user);
        dbContext.SaveChanges();
        return true;
    }
    return false;
}
```

## Login

Probiert der Benutzer sich mit seinen Logindaten anzumelden, so wird das eingegebene Passwort mit dem Salt (In der DB gespeichert) gehashed. Der Passworthash wird dann mit dem Passwort aus der Datenbank verglichen. Falls die beiden Passwörter übereinstimmen, ist der Benutzer erfolgreich authentifiziert.

Siehe Code Beispiel:

```
1 reference | SaraZarubica, 35 days ago | 1 author, 1 change | 0 exceptions
public bool Login(string userName, string password)
{
    bool result = false;
    User u = GetUserByUsername(userName);
    if(u != null)
    {
        byte[] typedPw = PasswordHelper.GenerateSaltedHash(StringUtils.GetByteArray(password), u.Salt); // eingegebenes pw beim login wird mit dem salt (der in der DB ist) gehasht
        result = PasswordHelper.CompareByteArrays(u.Password, typedPw); // pw der datenbank (schon beim registrieren gesalthashed) wird mit dem jetzigen hash verglichen, falls gleich user eingeloggt
    }

    return result;
}
```

## Aufgabe 1

Alle Hinweise und administrativen Vorgaben wurden umgesetzt.

## Aufgabe 2

Beschreiben Sie, auf welchem Tier Sie die dynamischen Elemente der Anwendung unterbringen möchten:

Tier 1 (Presentation):	Mit Javascript werden die Events (Button-Click, usw.) gehandelt. Mit HTML5 und CSS3 wird die Darstellung geregelt.
Tier 2 (Webserver):	Reagiert auf Events (z.B. Button-Clicks) und generiert die Web-Forms spezifischen Tags zu HTML5
Tier 3 (Applicationserver):	Hier wird die ganze Logik der Applikation behandelt. Z.B. wird geprüft ob die eingegebene Antwort Richtig war.
Tier 4 (Dataserver):	Mit dem EntityFramework (DataLayer, Code First) haben wir Zugang zum MSSQL Server. Das EntityFramework agiert hier als Datenbank Ersteller und als OR-Mapper. Mithilfe von LINQ to Entities (ADO.Net Entity Framework) machen wir die Abfragen auf dem MSSQL-Server.

## Aufgabe 3

Beschreiben Sie, welche Programmiersprache bzw. Technologien Sie verwenden möchten für:

Tier 1 (Presentation):	HTML 5, CSS, JavaScript
Tier 2 (Webserver):	asp.net WebForms (C#)
Tier 3 (Applicationserver):	C#
Tier 4 (Dataserver):	EntityFramework(Code First) C#, Datenbankserver MSSQL

## Aufgabe 4

Mit dem EntityFramework (DataLayer, Code First) habe ich(mithilfe des Connectionstrings im Web.config) Zugang zum MSSQL Server. Das EntityFramework agiert hier als Datenbank Ersteller und als OR-Mapper. Mithilfe von LINQ to Entities (ADO.Net Entity Framework) mache ich die Abfragen auf dem MSSQL-Server.

Die Abbildung auf eine Datenbank, steuern bestehende Klassen, welche mit Annotationen ausgezeichnet werden. Vom DbContext werden die Datenbank und die Datenbanktabellen modelliert und beim Aufruf von SaveChanges erstellt.

## Aufgabe 5

## Prototyp Quiz-Seite Kategorie wählen

[Wer wird Millionär?](#) [Über](#) [Kontakt](#) [Admin](#) [Log in](#)

### Wähle eine Spielkategorie

Allgemein,	▲
Mathematik ▶	○
Tier	▼

Tipp: Um mehrer Kategorien auszuwählen, halten Sie die Ctrl Taste gedrückt

[Spiel Starten](#)

## Prototyp Quiz-Seite

[Wer wird Millionär?](#) [Über](#) [Kontakt](#) [Admin](#) [Log in](#)

## Frage

[50:50](#)

[Antwort 1](#) [Antwort 2](#) [Punkte: 0](#)

[Antwort 3](#) [Antwort 4](#) [Frage wurde richtig beantwortet zu: 0%](#)

[Aufhören](#)

## Prototyp Quiz-Seite Name für Highscoreliste

[Wer wird Millionär?](#) [Über](#) [Kontakt](#) [Admin](#) [Log in](#)

Sie haben 1 Million gewonnen!

Geben Sie Ihren Namen für die Highscoreliste ein:

## Prototyp Quiz-Seite Highscoreliste

[Wer wird Millionär?](#) [Über](#) [Kontakt](#) [Admin](#) [Log in](#)

### Highscore

▼Rang	▼Gewichtete Punkte	▼Name	▼Zeitpunkt	▼Punkte	▼Dauer	▼Kategorien
1	11	Hans	05.04.2017 23:10:21	5	12 Sekunden	Auto
2	6	Martin	05.04.2017 23:10:21	5	14 Sekunden	Auto
3	5	Susi	05.04.2017 23:10:21	4	17 Sekunden	Auto
4	2	Tanja	05.04.2017 23:10:21	3	20 Sekunden	Auto

## Prototyp Admininterface Startseite

[Wer wird Millionär?](#) [Über](#) [Kontakt](#) [Admin](#) [Log out](#)

Was möchten Sie bearbeiten?

[Fragen](#) [Kategorien](#) [Highscore](#)

## Prototyp Admininterface Fragenliste

[Wer wird Millionär?](#) [Über](#) [Kontakt](#) [Admin](#) [Log out](#)

▼Frage	▼Kategorie
?	Auto
?	Auto
?	Auto
?	Auto

[Frage hinzufügen](#)

[Zurück](#)



## Prototyp Admininterface Fragenbearbeitung

The screenshot shows a web interface for editing a question. At the top, there is a dark navigation bar with links: "Wer wird Millionär?", "Über", "Kontakt", "Admin", and "Log out". The main content area is titled "Frage". It contains several input fields and dropdown menus. The "Kategorie" field has a dropdown menu with "Auto" selected. The "Frage" field is a text input with the placeholder text "Frage". Below this, there are three "Antwort" (Answer) sections. Each section has a text input for the answer and a dropdown menu for "Richtigkeit" (Correctness). The first two sections have their "Richtigkeit" dropdowns set to "Richtigkeit". The third section has a checkbox to its left. At the bottom of the form, there are three buttons: "Speichern" (Save), "Löschen" (Delete), and "Abbrechen" (Cancel).

Wer wird Millionär? Über Kontakt Admin Log out

Frage

Kategorie

Auto ▼

Frage

Frage

Antwort 1

Antwort

Richtigkeit ▼

Antwort 2

Antwort

Richtigkeit ▼

Antwort 3

Antwort

Richtigkeit ▼

☐

Antwort 4

Antwort

Richtigkeit ▼

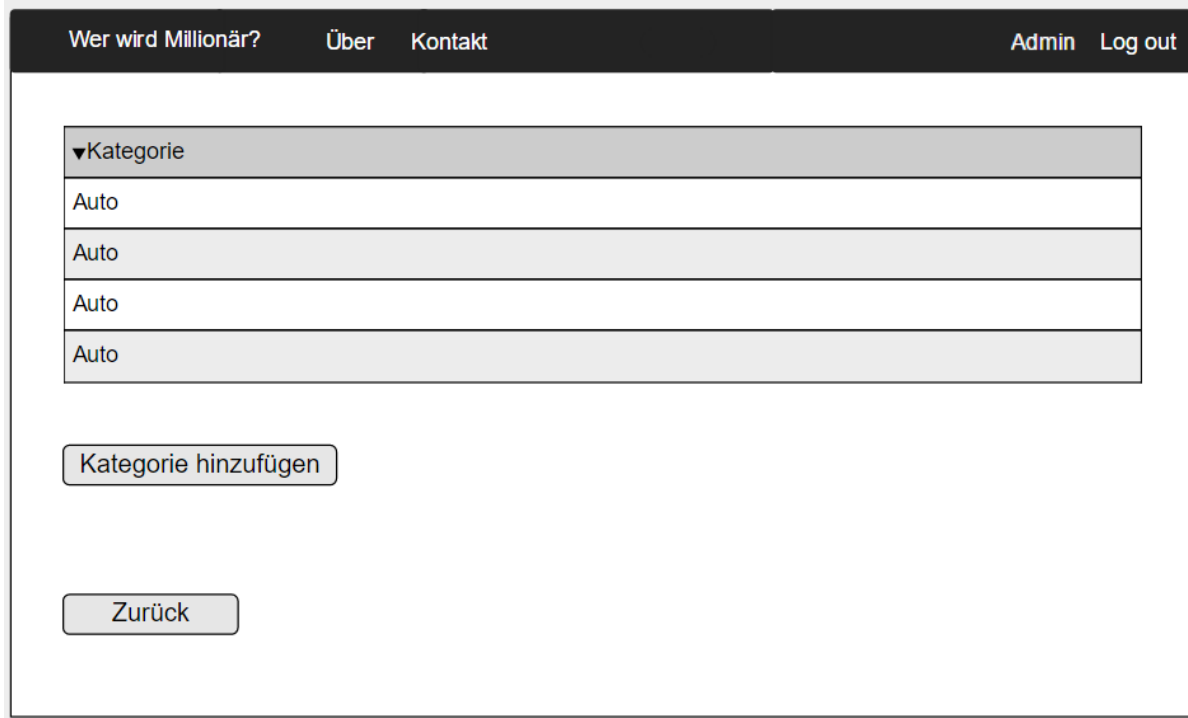
Speichern Löschen

Abbrechen

Hat man auf der Seite der Fragenliste «Frage hinzufügen» gedrückt, so kommt diese Seite (Siehe Bild) mit leeren Feldern.

Hat man jedoch in der Liste auf eine Frage gedrückt, so kommt die dementsprechende Frage ausgefüllt auf der neuen Seite und man kann sie bearbeiten oder löschen.

## Prototyp Admininterface Kategorienliste



Wer wird Millionär? Über Kontakt Admin Log out

▼Kategorie
Auto
Auto
Auto
Auto

Kategorie hinzufügen

Zurück

## Prototyp Admininterface Kategorienbearbeitung



Wer wird Millionär? Über Kontakt Admin Log out

Kategorie

Kategorie

Speichern Löschen

Abbrechen

Hier funktioniert es mit demselben Prinzip wie beim Frage Form(Siehe oben).

## Prototyp Admininterface Highscoreliste

Wer wird Millionär?
Über
Kontakt
Admin
Log out

▼Rang	▼Gewichtete Punkte	▼Name	▼Zeitpunkt	▼Punkte	▼Dauer	▼Kategorien
1	11	Hans	05.04.2017 23:10:21	5	12 Sekunden	Auto
2	6	Martin	05.04.2017 23:10:21	5	14 Sekunden	Auto
3	5	Susi	05.04.2017 23:10:21	4	17 Sekunden	Auto
4	2	Tanja	05.04.2017 23:10:21	3	20 Sekunden	Auto

Zurück

## Prototyp Admininterface Highscore löschen

Wer wird Millionär?
Über
Kontakt
Admin
Log out

### Highscore Eintrag

Rang: 4

Gewichtete Punkte: 5

Name Spieler: Anna

Zeitpunkt Spiel: 06.04.2017 12:46:57

Punkte: 60

Dauer Spiel: 12 Sekunden

Gewählte Kategorien: Auto

Löschen

Zurück

Klickt man in der Highscoreliste auf einen Eintrag, so wird dieser aufgelistet und man kann ihn löschen.

## Aufgabe 6

Die Webapplikation wurde nach den Vorgaben implementiert.

Überprüfungen:

1. Tier: Client

Mithilfe des HTML5 Tags «required» wird Clientseitig überprüft, ob das Feld ausgefüllt wurde. Da jedoch alles was beim Client läuft auch manipulierbar ist, werden diese Eingaben nochmals auf dem Server geprüft.

2. Tier: Webserver (ASP.Net WebForms [Code Behind])

Die einfachen und erweiterten Prüfungen der Eingaben (z.B. Validierung einer Emailadresse) finden auf dem Server statt, da erweiterte Prüfungen auf dem Client einen zusätzlichen Aufwand bedeutet hätten und zudem sehr leicht manipulierbar wären (Bearbeitung des Javascripts beim Client).

## Aufgabe 7

Alle notwendigen Daten für die Abnahme befinden sich unter dem Punkt «Allgemein» im Inhaltsverzeichnis. Das Abnahmeprotokoll ist hier aufgelistet.

### Abnahmeprotokoll

Auftraggeber:	Berufsfachschule Baden Wiesenstrasse 32 5400 Baden 056 222 02 06 sekretariat@bbb Baden.ch  Manuel Bachofner	Auftragnehmer:	Sara Informatik Zelgmattstrasse 14 8956 Killwangen 076 464 1306 sara13@hotmail.ch  Sara Zarubica
Projektbezeichnung:	Es soll ein «Wer wird Millionär» implementiert werden. In der Webapplikation soll die Frageform – eine Frage mit vier Antwortmöglichkeiten, von denen nur eine korrekt ist – beibehalten werden. Für jede korrekt beantwortete Frage, erhält der Spieler 30 Punkte. Damit die Recherchemöglichkeit eingeschränkt wird, soll die Zeit zwischen dem Start des Quiz und dem Ende gemessen werden. Der 50:50 Joker soll dem Spieler einmal zur Verfügung stehen. Benutzt er ihn, werden zwei falsche Antworten ausgeblendet. Die Fragen werden von einem Administrator gepflegt und unterhalten. Als Ansporn gibt es eine Highscoreliste, die den Vergleich der Resultate ermöglicht.		
Projektbeginn:	07.02.2017	Projektende:	09.04.2017
Datum:	04.04.2017	Ort:	Baden

## Anforderungen

Punkt:	Beschreibung	+	-
1	Der Administrator kann sich mit Benutzernamen und Passwort anmelden.	X	
2	Der Administrator kann, Fragen anlegen, ändern und löschen.	X	
3	Der Administrator kann, Kategorien anlegen und einer Frage zuordnen.	X	
4	Der Administrator kann einzelne Einträge der Highscoreliste löschen.	X	
5	Als Client dient ein Webbrowser.	X	
6	Zu jeder Frage wird gespeichert, wie oft sie richtig und wie oft sie falsch beantwortet worden ist.	X	
7	Zu jeder Frage wird beim Quiz prozentual angezeigt, wie oft sie richtig beantwortet wurde.	X	
8	Der Spieler sieht zu jeder Zeit seine aktuelle Punktzahl.	X	
9	Der Spieler sieht zu jeder Zeit, ob er den 50:50 Joker noch einsetzen kann.	X	
10	Der 50:50 Joker markiert zwei falsche Antworten und macht sie unauswählbar.	X	
11	War die vom Spieler gewählte Antwort richtig, so wird dies dem Spieler mitgeteilt und weiter zur nächsten Frage gegangen.	X	
12	War die vom Spieler gewählte Antwort falsch, so wird dies dem Spieler mitgeteilt und die richtige Antwort dargestellt.	X	
13	Hat der Spieler eine falsche Antwort eingegeben, so bricht das Quiz ab, der Versuch wird mit 0 Punkten gewertet und erscheint nicht auf der Highscoreliste, ansonsten schon.	X	
14	Der Spieler muss seinen Namen angeben können, mit dem er auf der Highscoreliste erscheint.	X	
15	In der Highscoreliste werden folgende Daten aufgeführt: Rang, Gewichtete Punkte(Punktzahl / Dauer des Quiz in Sekunden), Name des Spielers, Zeitpunkt des Spielers, Anzahl Punkte, Dauer des Quiz und gewählte Kategorien.	X	
16	Die Highscoreliste wird nach Rang, der durch die gewichteten Punkte bestimmt wird, aufsteigend sortiert.	X	
17	Der Spieler kann die Kategorien wählen, aus denen die Fragen zufällig ausgewählt werden.	X	
18	Die Fragen mit den Antworten werden dem Spieler nacheinander präsentiert.	X	
19	Keine Frage soll während eines Spieles mehr als einmal gestellt werden.	X	
20	Zu jeder Frage gibt es vier Antworten: Eine korrekte und drei falsche.	X	
21	Der Spieler kann entweder eine Antwort auswählen oder aufhören und seinen Gewinn realisieren.	X	

22	Jede korrekt gewählte Antwort gibt 30 Punkte auf das Spielerkonto.	X	
23	Das Spiel soll mit einer spielbaren Anzahl Fragen gefüllt werden.	X	
24	Die Zeit zwischen dem Start des Quiz und dem Aufhören soll gemessen werden.	X	
25	Einfache Formulareingaben, wie leere Textfelder etc., sollen auf Client- und Serverseite geprüft werden.	X	
26	Transaktionsmanagement ist einzusetzen.	X	
27	Sicherheitsaspekte werden umgesetzt.	X	
28	Die Applikation soll als sessionbasierte, professionelle 4-Tier Architektur implementiert werden.	X	
Mängel			
Nummer:	Punkt:	Beschreibung:	Frist:
Keine Mängel			
Bemerkungen:	Das Projekt wurde mit Selenium getestet. Unit Tests wurden durchgeführt.		
Vorbehalte:	Keine		
Unterschrift/Datum Auftraggeber	Manuel Bachofner	Unterschrift/Datum Auftragnehmer	Sara Zarubica 06.04.2017

## Aufgabe 8

### Testumgebung

Die Tests wurden mit Selenium in Mozilla Firefox durchgeführt.

### Testfallspezifikation

Testfall-Nr. 1	
Anforderung:	Bei ungültiger Eingabe der Logindaten wird eine Fehlermeldung angezeigt.
Voraussetzung:	Login Seite geöffnet.
Eingabe:	<ol style="list-style-type: none"> <li>1. Username: Test</li> <li>2. Passwort: Test</li> </ol>
Ausgabe:	Fehlermeldung «Ungültige Benutzerdaten» wird angezeigt.

Testfall-Nr. 2	
Anforderung:	Bei gültiger Eingabe der Logindaten wird der Nutzer auf die Adminseite weitergeleitet.
Voraussetzung:	Login Seite geöffnet.
Eingabe:	<ol style="list-style-type: none"> <li>1. Username: admin</li> <li>2. Passwort: admin1234</li> </ol>
Ausgabe:	Weiterleitung auf die Adminseite.

Testfall-Nr. 3	
Anforderung:	Der Admin kann eine Frage erstellen.
Voraussetzung:	Fragenliste Seite geöffnet.
Eingabe:	<ol style="list-style-type: none"> <li>1. Button «Frage hinzufügen» betätigen</li> <li>2. Ausfüllen der Seite /QuestionEdit</li> <li>3. Button «Speichern» betätigen</li> </ol>
Ausgabe:	<p>Weiterleitung auf /QuestionEdit.</p> <p>Das Formular zur Eingabe wird angezeigt.</p> <p>Nach Speicherung wird die ganze Liste mit den Fragen angezeigt.</p>



Testfall-Nr. 4	
Anforderung:	Der Admin kann eine Frage löschen.
Voraussetzung:	Fragenliste Seite geöffnet.
Eingabe:	<ol style="list-style-type: none"> <li>1. Klick auf die gewählte Frage in der Liste(/QuestionList).</li> <li>2. Betätigen des Button «Löschen» in /QuestionEdit.</li> </ol>
Ausgabe:	Die Liste der Fragen(/QuestionList) wird angezeigt und die gelöschte Frage ist nicht mehr in der Liste vorhanden.

Testfall-Nr. 5	
Anforderung:	Der Admin kann eine Frage ändern.
Voraussetzung:	Fragenliste Seite geöffnet.
Eingabe:	<ol style="list-style-type: none"> <li>1. Klick auf die gewählte Frage der Liste in /QuestionList.</li> <li>2. Gewünschte Änderung vollbringen in /QuestionEdit.</li> <li>3. Button «Speichern» betätigen.</li> </ol>
Ausgabe:	Die Liste der Fragen wird angezeigt und die bearbeitete Frage ist geändert.

Testfall-Nr. 6	
Anforderung:	Der Admin kann eine Kategorie erstellen.
Voraussetzung:	Kategorienliste Seite geöffnet.
Eingabe:	<ol style="list-style-type: none"> <li>1. Button «Kategorie hinzufügen» betätigen</li> <li>2. Eingabe der Kategorie auf /CategoryEdit.</li> <li>3. Button «Speichern» betätigen</li> </ol>
Ausgabe:	Die Liste mit den Kategorien(/CategoryList) wird angezeigt und die neue Kategorie ist aufgelistet.

Testfall-Nr. 7	
Anforderung:	Admin kann Highscore Einträge löschen
Voraussetzung:	Highscoreliste Seite geöffnet.
Eingabe:	<ol style="list-style-type: none"> <li>1. Klick auf gewählten Highscore Eintrag</li> <li>2. Auf /HighscoreEdit den Button «Löschen» betätigen</li> </ol>
Ausgabe:	Die Liste der Highscore Einträge(/HighscoreList) wird ohne das gelöschte Objekt angezeigt.

Testfall-Nr. 8	
Anforderung:	Eine Fehlermeldung soll angezeigt werden, wenn die gewählte Kategorie/n zu wenig Fragen haben.
Voraussetzung:	/PlayerChooseCategory Seite geöffnet.
Eingabe:	<ol style="list-style-type: none"> <li>1. Klick auf gewählte Kategorie/n</li> <li>2. Button "Spiel starten" betätigen</li> </ol>
Ausgabe:	Die Fehlermeldung mit dem Text "Die gewählte Kategorie/en hat nicht genügend Fragen. Wählen sie noch zusätzliche Kategorien an, oder informieren Sie den Admin." wird angezeigt.

Testfall-Nr. 9	
Anforderung:	Das Spiel soll beginnen, wenn die gewählte Kategorie/n genügend Fragen hat.
Voraussetzung:	/ PlayerChooseCategory Seite geöffnet.
Eingabe:	<ol style="list-style-type: none"> <li>1. Klick auf gewählte Kategorie/n</li> <li>2. Button "Spiel starten" betätigen</li> </ol>
Ausgabe:	Das Spiel startet.

Testfall-Nr. 10	
Anforderung:	Mehrere Kategorien können gewählt werden.
Voraussetzung:	/ PlayerChooseCategory geöffnet.
Eingabe:	Klick auf gewählte Kategorien Button "Spiel starten" betätigen
Ausgabe:	Das Spiel wird gestartet und es sind mehrere Kategorien vorhanden.

Testfall-Nr. 11	
Anforderung:	Der 50:50 kann angewendet werden.
Voraussetzung:	/Game Seite geöffnet.
Eingabe:	1. Button "50:50" betätigen
Ausgabe:	2 Fragen werden in den Hintergrund gesetzt und man kann nur noch 2 von 4 Fragen auswählen.

Testfall-Nr. 12	
Anforderung:	Bei richtiger Antwort wird die Antwort grün dargestellt.
Voraussetzung:	/ Game Seite geöffnet.
Eingabe:	1. Klick auf die richtige Antwort
Ausgabe:	Die Antwort leuchtet grün und nach einigen Sekunden kommt die nächste Frage.

## Testfall-Nr. 13

Anforderung:	Bei falscher Antwort wird die Antwort grün dargestellt.
Voraussetzung:	/Game Seite geöffnet.
Eingabe:	1. Klick auf die falsche Antwort.
Ausgabe:	Die Antwort leuchtet rot und die richtige grün. Nach einigen Sekunden kommt, dass wir das Spiel verloren haben.

## Testfall-Nr. 14

Anforderung:	Man kann das Spiel aufhören.
Voraussetzung:	/Game Seite geöffnet.
Eingabe:	1. Button "Aufhören" betätigen
Ausgabe:	Die Eingabe zum Highscore Eintrag erscheint, das Spiel wird beendet.

## Testfall-Nr. 15

Anforderung:	Man kann sich in die Highscoreliste eintragen.
Voraussetzung:	/PlayerWinOrStop Seite geöffnet.
Eingabe:	1. Eintrag des Namens in das Namenfeld 2. Button "Bestätigen" betätigen
Ausgabe:	Die Liste der Highscore Einträge wird angezeigt. Der eigen Highscore ist in der Liste ersichtlich.

## Testfall-Nr. 16

Anforderung:	Leere Eingabe bei Highscorelisten Eintrag wird abgefangen.
Voraussetzung:	/ PlayerWinOrStop Seite geöffnet.
Eingabe:	1. Leerer Eintrag in das Namenfeld. 2. Button "Bestätigen" betätigen
Ausgabe:	Die Fehlermeldung mit dem Text "Geben Sie Ihren Namen ein." wird angezeigt.

## Testfall Protokoll

Testnummer	Wer	Datum	Resultat	Bemerkung
1	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
2	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
3	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
4	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
5	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
6	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
7	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
8	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
9	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
10	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
11	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
12	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
13	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
14	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
15	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet
16	Sara Zarubica	06.04.2017	Bestanden	Mit Selenium getestet

## Fazit der Tests

Die Tests wurden alle erfolgreich ausgeführt.

## Aufgabe 9

Keine Erweiterung.