# Dokumentation WPF – Moderne Webanwendungen

## Webseite „Zocker Wonderland“

## von Jessica Lee Schulz, Jasmin Schermuly und Enrico Gette

Auf Webseite „Zocker Wonderland“ können die Benutzer nach Spielen suchen und zu dennen Links zu Tipps und Youtube-Videos speichern. Außerdem können Benutzer Spiele nach Konsolen filtern.

1. **PHP Klasse:**

Die Klasse *Datenbank* in der Datei db.php beinhaltet die Funktion „verbindung()“ welche für den Verbindungsaufbau der Datenbank zuständig ist. Die Verbindung zur Datenbank „ZockerWonderland“ wird mittels $mysqli erzeugt.

1. **SQL-Tabellen:**

Die benötigten Tabellen für die Datenbank sind in „sql/games.sql“ und „sql/login-datenbank.sql“ zu finden.

Zusätzliche Links zu jedem einzelnen Spiel werden in der „games“ - Tabelle abgespeichert.

Die Benutzerdaten wie Name, Email-Adresse und Passwort werden in der „users“ - Tabelle gespeichert.

1. **PHP - Session:**

Wenn sich jemand auf der Seite registrieren möchte, wird die Seite „registrieren.php“ geladen.

Diese enthält ein Formular, welches überprüft ob eine gültige Email-Adresse eingeben wurde, sowie dass nur ein Account pro Email-Adresse angemeldet werden kann. Des Weiteren wird überprüft ob das Passwort zwei Mal richtig angegeben wurde. Das Passwort wird nur als Hash in der Datenbank abgespeichert.

Nach der Registrierung kann der Benutzer sich über eine Eingabemaske einloggen. Diese Eingabemaske befindet sich in der login.php. Nach Überprüfung des Passwortes wird in Zeile 19 die Session über eine „userid“ angelegt.

Die Abmeldung erfolgt in der „logout.php“ mittles „session\_destory()“.

Eingeloggte Benutzer können zu einzelnen Spielen zusätzliche Links in der Datenbank abspeichern.

1. **AJAX- Request und js-Funktionen:**

Für die Webseite haben wir uns für die Giantbomb API (<http://www.giantbomb.com/api/documentation> ) entschieden. Alle Funktionen für den AJAX-Request befinden sich in der „/js/api2.js“.

Die Funktion „search()“ in wird entweder über den Klick auf den „suche“-Button oder über die Enter-Taste gestartet. Die Eingabe wird in der Variable „query“ gespeichert und danach wird die Funktion „sendRequest(resource, filter2, data)“ aufgerufen.

Die Funktion „getPlatformGames(id)“ dient für eine Anfrage an die API, gefiltert nach einer spezifischen Konsole, welche über eine ID angesprochen wird. Hier wird ebenfalls für den Request die Funktion „sendRequest“ aufgerufen.

Die Funktion „getGame(gameID)“ dient für eine Anfrage an die API, um Informationen zu einem einzelnen Spiel zu erhalten. . Hier wird ebenfalls für den Request die Funktion „sendRequest“ aufgerufen.

Die Funktion „sendRequest(resource, filter2, data)“ dient zum eigentlichen AJAX-Request auf die Giantbomb API. In der Funktion wird zuerst in der Variable „apiKey“ unser Key für den Zugang zur API gespeichert. Für jeden Request muss eine spezifische Request-URL aufgebaut werden die sich aus der BasisURL, der Resource, den Filtern sowie unserem API- Key zusammenstellt. Da auf unserer Seite entweder eine Liste von Spielen oder Informationen zu einzelnen Spielen angezeigt werden, mussten zwei verschiedene Callback-Funktionen angegeben werden. Nun kann der AJAX-Request abgeschickt werden.

Die Callback Funktion „showResults(result)“ filtert die Daten der API nach geeigneten Informationen zu jedem einzelnen Spiel formatiert diese Informationen in HTML-Tags, welche gemeinsam in ein „div“ mit der ID „result“ geladen werden. Genauso verhält es sich bei der Callback Funktion „showResultsgame(result)“, nur dass hierbei die Informationen der API eines spezifischen Spiels gefiltert werden.

1. **Genereller Aufbau der Webseite:**

Aufbau der „Index.php“: Include von „header.php“, „navi.php“, „content.php“, „footer.php“.

Aufbau der “header.php”: HTML-Meta-Informationen, Einbindung von „css/style.css“ und „/js/api2.js“, Überprüfung ob der Benutzer eingeloggt ist mittels PHP-Session, include von „login.php“ und „logout.php“

Aufbau der „game.php“: Abfrage der Datenbank nach zusätzlichen Informationen zu einem Spiel, sowie die Darstellung dieser Informationen.