



Projektdokumentation

Modul 151 - Projektarbeit

Natascha Wernli & Sara Roth

Inhaltsverzeichnis

Management Summary.....	2
Ausgangslage.....	2
Umsetzung	2
Ergebnis.....	2
Einleitung	3
technische Umsetzung	4
Projektantrag erstellen.....	4
Projektdokumentation	4
Usability mit sinnvollen Meldungen	4
Client- und serverseitige Validierung	4
Sessionhandling.....	4
Sichere Speicherung von Daten.....	5
SQL-Injection, Script-Injection und Session-Hijacking werden verhindert.....	5
Registrierung und Login.....	5
Daten Erfassen Ändern und Löschen.....	5
Testplan erstellen und relevante Funktionen prüfen	5
Validierung von HTML und CSS	5
Planungstool.....	5
strukturierter Code.....	6
kommentierter Code.....	6

Management Summary

Ausgangslage

Alle Noten der Schüler wurden bis vor kurzen von den Lehrerinnen in eine eigene Excel Tabelle eingetragen und am Ende vom Semester wurde es mit den anderen Tabellen der Lehrer ins Zeugnis übernommen, dass den Schülern abgegeben wurde. Die Schüler hatten also keinen Überblick über die eigenen Noten und haben sie immer erst durch das Zeugnis mitgeteilt bekommen. Nun hat die Schüler Geld zur Verfügung gestellt bekommen, um Webentwickler anzustellen, die diese ganzen Administrationen digital umsetzen können. Das Erste, das digitalisiert wurde, war diese Notenverwaltung. Es war klar, dass diese so nicht weitergeführt werden kann.

Umsetzung

Um dieses Problem mit der Notenverwaltung zu lösen und übersichtlicher zu machen, wurde ein Projekt gestartet, indem eine neue Web-Applikation entstehen sollte. Die Applikation wurde grösstenteils mit PHP, HTML und CSS aufgebaut. Zu einem kleinen Teil wurde zusätzlich auch noch Java Script eingesetzt. Das Projekt wurde nach der Planung in zwei Sprints aufgeteilt. Ein Sprint dauerte etwa zwei Wochen an, indem die zugeteilten Tickets umgesetzt werden mussten.

Ergebnis

Dank der erfolgreichen Umsetzung von diesem Projekt können nun alle Schüler, die Noten zu jeder Zeit nachschauen gehen. Alle Lehrer sind nun in der Lage, die Noten nach einer Prüfung in die Applikation einzutragen, damit die Schüler immer auf dem neusten Stand sind. Einem Benutzer mit Adminrechten ist es jederzeit möglich, einen neuen Benutzer im System zu erstellen. Ebenso kann ein User auch gelöscht werden.

Einleitung

Die Projektdokumentation ist eine Zusammenstellung von ausgewählten Daten. In der Projektdokumentation stehen Informationen zur Planung und Umsetzung der Web-Applikation zur Verfügung. Somit ist sie auch ein Hauptbestandteil dieses Projekts.

In diesem Projekt wird die Excel Tabelle zur Verwaltung von den Noten, die die Schüler schreiben, in eine Web-Applikation verpackt. Es sollte die Handhabung von beiden Seiten aus verbessern.

Bevor die Applikation zur Abgabe für die Produktion gegangen ist, wurde sie auf funktionale Schwerpunkte getestet und wenn nötig noch überarbeitet.

technische Umsetzung

Projektantrag erstellen

Der erste Schritt von unserem Projekt war der Projektantrag. Dazu wurde ein Dokument erstellt, indem unsere Idee preisgegeben wurde und viele wichtige Komponenten dazu dokumentiert wurden. Anhand des Projektantrages kennt man die Problembeschreibung und wie es zum Projekt kam, die Technologien, die verwendet werden, die Personen, die daran beteiligt sind und die grobe Planung wer was macht usw. Ein Projektantrag bildet die Basis und auch den Start eines Projekts. Nach einer Genehmigung kann mit dem Umsetzen offiziell gestartet werden.

Projektdokumentation

In diesem Dokument werden alle Funktionen erklärt und beschrieben die in der Applikation vorhanden sind. Diese Dokumentation hilft uns für die Widerspiegelung aller Kompetenzen, die erreicht werden müssen und geben einen guten Überblick über die gesamte Applikation und allen implementierten Funktionen.

Usability mit sinnvollen Meldungen

GreatGrade ist so aufgebaut, dass der Benutzer der Applikation sich gut zurechtfindet. Es fällt ihm leicht, die Applikation zu bedienen und er kann all seine Berechtigungen, die er hat, anwenden. Bei Benutzer-Inputs wird er mit entsprechenden Meldungen durch den Prozess geführt, damit er bei Interaktionen mit der Applikation informiert wird.

Client- und serverseitige Validierung

Die Benutzereingaben werden client- und serverseitig validiert, um zu verhindern, dass falsche Daten in der Datenbank gespeichert werden können. Die Daten, die der Benutzer im Login eingibt, werden absichtlich nur grob validiert, damit ein Benutzer ohne Login nicht zu viele Informationen über die Formate erhalten kann. Beim Login wird nur überprüft, ob die Benutzereingaben mit den Einträgen in der Datenbank auch wirklich übereinstimmen. Serverseitig wird natürlich überprüft, ob die Felder einen Wert haben und diesen Wert wird wiederum getrimmt und mit `htmlspecialchars()` validiert.

Sessionhandling

Um die Kommunikation von Webapplikation und dem Benutzer unter Kontrolle zu haben wird Sessionhandling eingesetzt. Wenn der Benutzer eine Session gestartet hat kann er die Applikation mit den zur Verfügung gestellten Funktionen gebrauchen. Es ist nicht nur möglich das Passwort zu ändern, sondern er kann sich auch zu jederzeit ausloggen und die Session wieder zu beenden.

Der benutzer wird mit einer Error-Page aufmerksam gemacht, dass er noch keine Session gestartet hat und diese wiederum leitet ihn auf die Login Seite weiter, um eine Session zu starten.

Sichere Speicherung von Daten

Dem Datenbankbenutzer wurden nur die nötigsten Rechte vergeben, die für die Kommunikation zwischen der Datenbank und der Web-Applikation benötigt werden. Im Script für die Datenbank (script_database_gg.sql) wird dazu gleich auch noch einen Datenbankbenutzer erstellt. Diesem Benutzer wurden nur die Rechte select, update, insert und delete gegeben.

Da Passwörter besonders sicher gespeichert werden sollen werden die Passwörter mit einem Hash-Wert plus einem Salt Zusatz an die Datenbank geschickt.

SQL-Injection, Script-Injection und Session-Hijacking werden verhindert

Im gesamten Code wird SQL-Injection, Script-Injection und Session-Hijacking verhindert. Dies wurde mit den verschiedensten Funktionen, die PHP zur Verfügung stellt und die wir im Unterricht kennengelernt haben umgesetzt. Damit sollte verhindert werden, dass falsche Daten an den Server oder in die Datenbank gesendet werden. Es wird auch verhindert, dass eine Session mitgehört werden kann und die ID bekannt ist.

Registrierung und Login

Der Admin ist in der Lage bei den Einstellungen auf der Webseite einen neuen Benutzer zu registrieren. Dabei werden Daten angegeben, die später für das Login gebraucht werden, Username und Passwort, sowie den Namen angegeben. Weitere Daten über einen Benutzer werden zurzeit für die Applikation nicht gebraucht. Bevor ein Benutzer Zugriff auf die Applikation bekommen kann, muss er sich in der Login-Maske mit den korrekten Daten anmelden.

Daten Erfassen Ändern und Löschen

In unserem Projekt kann einem Benutzer eines von drei verschiedenen Rechten zugeteilt werden. Jeden Benutzer wird ermöglicht, sein Passwort jederzeit ändern zu können. Ein Benutzer, der Adminrechte besitzt, ist in der Lage, ein neuer Benutzer zu erstellen oder ein bereits bestehender Benutzer zu löschen. All diese Änderungen werden in die Datenbank geschrieben und übernommen.

Testplan erstellen und relevante Funktionen prüfen

Der Testplan wurde erstellt und alle Tests wurden durchgeführt. Diese wurden wiederum kurz dokumentiert. Der Testplan mit allen Testfällen wurde auf einem separaten Dokument geschrieben. Dieser wurde mit dem gesamten Projektordern mitgeschickt und ist im Ordner "Dokumente" zu finden.

Validierung von HTML und CSS

Unser Projektcode, ausschliesslich PHP und JS wurde mit der Webseite <https://validator.w3.org/> validiert.

Planungstool

Als Planungstool wurde Jira gewählt. Wir haben uns für Jira entschieden, weil das Projektmanagement sehr einfach und strukturiert aufgebaut wird. Ein weiterer Grund, wieso wir uns dafür entschieden haben ist, dass wir im ÜK oder aber auch auf geschäftlicher Basis damit arbeiten. Das Prinzip ist agil aufgebaut und man kann die Tasks in sogenannte Sprint's einteilen. Ein Sprint hat eine genaue Zeitspanne definiert, in der die Tickets möglichst bearbeitet werden sollen.

strukturierter Code

Der Programmcode wurde strukturiert aufgebaut. PHP und HTML Code wurden in der Datei grösstenteils getrennt. Dies führt zu einer besseren Übersicht.

kommentierter Code

Der Code wurde ausführlich an allen relevanten Stellen kommentiert. Auch wenn die Kommentare nur sehr kurz sind, ist trotzdem klar, was im Code genau passiert. Eine dritte Person sollte somit in der Lage sein, den Code mit den Kommentaren verstehen zu können.