Projektdokumentation

Entwickeln einer Website für eine Pfadfindergruppe

Autoren: Fabian Hasse

Maximilian Dierkes

Version: 1.0

Abgabedatum: 05.02.2016

Inhalt

[1. Projektbeschreibung 2](#_Toc444006694)

[1.1 Ausgangslage und Projektumfeld 2](#_Toc444006695)

[1.2 Beschreibung Ist-Zustand 2](#_Toc444006696)

[1.3 Beschreibung Soll-Zustand 2](#_Toc444006697)

[1.4 Systemumgebung 2](#_Toc444006698)

[1.4.1 Hardware 2](#_Toc444006699)

[1.4.2 Software 3](#_Toc444006700)

[1.4.3 Organisation und Hilfsmittel 3](#_Toc444006701)

[2. Projektübersicht 3](#_Toc444006702)

[2.1 Projektplan 3](#_Toc444006703)

[2.2 Kostenplanung 3](#_Toc444006704)

[2.2.1 Personalkosten 3](#_Toc444006705)

[2.2.2 Materialkosten 4](#_Toc444006706)

[2.2.2.1 Hardwarekosten 4](#_Toc444006707)

[2.2.2.2 Softwarekosten 5](#_Toc444006708)

[2.2.3 Sonstige Kosten 5](#_Toc444006709)

[2.2.4 Gesamtkosten 5](#_Toc444006710)

[3. Durchführung und Protokoll des Projektverlaufs 6](#_Toc444006711)

[3.1 Ist-Zustands-Analyse 6](#_Toc444006712)

[3.2 Konzeption des späteren Seitenlayouts 6](#_Toc444006713)

[3.3 Erstellung des Pflichtenhefts 6](#_Toc444006714)

[3.4 Datenbankentwurf 7](#_Toc444006715)

[3.5 Implementierung 8](#_Toc444006716)

[3.6 Testphase 9](#_Toc444006717)

[3.7 Erstellung der Projektdokumentation 9](#_Toc444006718)

# Projektbeschreibung

## Ausgangslage und Projektumfeld

Das im folgenden beschriebene Projekt ist als Teil der Ausbildung zum Fachinformatiker für Anwendungsentwicklung bei der dSPACE GmbH anzusehen.  
Das Projekt wird für die Pfadfindergruppe der DPSG St. Meinolf Paderborn erstellt, die somit der Auftraggeber des Projekts ist. Durchgeführt wurde das Projekt in den Räumlichkeiten des b.i.b.

## Beschreibung Ist-Zustand

Die DPSG St. Meinolf ist ein seit über 65 Jahre in Paderborn ansässiger Pfadfinderstamm mit über 100 aktiven Mitgliedern.   
Die aktuelle Webseite der Pfadfindergruppe sieht veraltet aus, ist nicht besonders benutzerfreundlich und nicht für mobile Geräte geeignet.  
Es ist schwierig neue Newseinträge zu verfassen und unter anderem Gruppenseiten zu bearbeiten, da dies nur über eine Person geschieht. Dies führt zu wenigen neuen Inhalten und Gruppenseiten mit veralteten Informationen. Zudem sind die Newsartikel nur aus einer Perspektive beschrieben, was die Vielfältigkeit in einem Pfadfinderstamm nicht realitätsgetreu wiederspiegelt.

## Beschreibung Soll-Zustand

Ziel des Projekts ist eine Webseite mit modernem Aussehen, die sowohl auf Desktops als auch mobil sehr gut funktioniert.  
Um die im Ist-Zustand beschriebenen Probleme zu lösen, sollen folgende Funktionen umgesetzt werden.  
Zum einen soll es möglich sein neue Newseinträge erstellen, bearbeiten und löschen zu können. Damit dies nicht nur einer Person zugänglich ist, soll eine Benutzerverwaltung mit Rechtesystem implementiert werden.  
Auf der Webseite soll die Möglichkeit geboten werden, sich zu jedem Zeitpunkt mit einem Nutzernamen und Passwort einzuloggen, um Newseinträge verfassen zu können. Zudem können eingeloggte Nutzer vorhandene Newseinträge kommentieren.  
Des weiteren haben einige Nutzer die Möglichkeit, die Gruppenseiten bearbeiten zu können.  
All diese Funktionen sollen auch auf mobilen Endgeräten, wie Smartphones oder Tablets, verfügbar und leicht zu bedienen sein.

## Systemumgebung

## Hardware

Die in dem Projekt zu erstellende Webseite soll serverseitig auf einem NodeJS Server basieren.  
Auf der Clientseite wird ein Endgerät erwartet, welches die Webseite mit dem Browser Google Chrome in mindestens Version 40 oder dem Browser Firefox in Version 35 oder höher aufruft.  
Auf der Serverseite muss ein System mit den folgenden Leistungsmerkmalen mindestens gegeben sein:

* 1,00 GB Arbeitsspeicher
* Dual Core CPU mit 1,00 GHz
* Linux basiertes Betriebssystem
* Netzwerkkarte mit Zugang zum Internet

## Software

Für die korrekte Darstellung der Webseite wird auf dem Client Endgerät mindestens der Browser Google Chrome Version 40 oder Mozilla Firefox in Version 35 benötigt. Zudem muss JavaScript in den Browsereinstellungen aktiviert sein.  
Auf dem Server läuft NodeJS in Version 5.5 mit verschiedenen Nodemodulen. Zudem existiert auf dem Server MySQL in Version 5.5 oder höher.   
Die Webseite basiert auf folgenden Open Source Libraries: Angular 2, Bootstrap, jQuery, SASS.

## Organisation und Hilfsmittel

Ansprechpartner für dieses Projekt, seitens des Auftraggebers DPSG St. Meinolf, ist Peter Stöber. Er wird das Projekt nach der Durchführung abnehmen und steht für inhaltliche Rückfragen zur Verfügung.  
Für fachliche Fragen ist, innerhalb des b.i.b., Herr Schlichting zuständig. Er ist gleichzeitig auch der Projektverantwortliche, der uns die Verantwortung zur Durchführung des Projekts erteilt hat.

Für die Arbeit an dem Projekt stehen uns zwei MacBooks mit Zugang zum b.i.b. Netzwerk und Internet, GitHub zur zentralen Source Code Verwaltung und das Textverarbeitungsprogramm Microsoft Word 2013 zur Erstellung von dem Pflichtenheft und der Projektdokumentation zur Verfügung.

# Projektübersicht

## Projektplan

Für das Projekt sind insgesamt 100 Schulstunden (2 x 50 Schulstunden) a 45 Minuten im Zeitraum vom 02.02.2016 bis zum 09.03.2016 wie folgt geplant:

|  |  |
| --- | --- |
| **Tätigkeit** | **Schulstunden** |
|  |  |
| Ist-Zustands-Analyse | 2 |
| Konzeption des späteren Seitenlayouts | 10 |
| Erstellung des Pflichtenhefts | 10 |
| Datenbankentwurf | 5 |
| Implementierung | 50 |
| Testphase | 5 |
| Erstellung der Projektdokumentation | 18 |
|  |  |
| ***Gesamtstunden in Schulstunden:*** | ***100*** |

## Kostenplanung

Für das durchzuführende Projekt werden verschiedene Kosten in der Kostenplanung aufgezeigt. Diese werden für die Durchführungszeit von 75 Stunden bzw. 100 Schulstunden anteilig berechnet.

## Personalkosten

Die bearbeitenden Mitarbeiter sind zwei Auszubildende der dSPACE GmbH im zweiten Lehrjahr.  
Zudem fallen Projektkosten für weitere Mitarbeiter, die zur Beratung hinzugezogen wurden, an.  
Die Personalkosten eines Auszubildenden setzen sich pro Jahr wie folgt für den Arbeitgeber zusammen:

|  |  |
| --- | --- |
| **Beschreibung** | **Kosten** |
| Ausbildungsvergütung pro Jahr (brutto) | 11.570,00 € |
| Arbeitgeberanteil Krankenversicherung (7,3%) | 844,61 € |
| Arbeitgeberanteil Pflegeversicherung (1,175%) | 135,95 € |
| Arbeitgeberanteil Arbeitslosenversicherung (1,5%) | 173,55 € |
| Arbeitgeberanteil Rentenversicherung (9,8%) | 1133,86 € |
| Unfallversicherung (0,33%) | 38,18 € |
|  |  |
| **Gesamtkosten des Arbeitgebers pro Jahr** | **13896,15 €** |

Bei einer täglichen Arbeitszeit von 8 Stunden pro Tag und den üblichen Werten von 230 Arbeitstagen pro Jahr, ergibt sich eine **Jahresstundenzahl** **von** **1840 Stunden**.

Daraus resultiert ein Stundenlohn für einen Auszubildenden im 2. Lehrjahr von:

**Personalkosten gesamt**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Person** | **Aufgabenbeschreibung** | **Dauer in Std.** | **Stundenlohn** | **Kosten** |
| Herr Schlichting | Rücksprache während des Projekts | 3,0 | 80,00 € | 240,00 € |
| Herr Dierkes | Durchführung des Projekts | 37,5 | 7,55 € | 283,13 € |
| Herr Hasse | Durchführung des Projekts | 37,5 | 7,55 € | 283,13 € |
|  |  |  |  |  |
| **Personalkosten gesamt:** | | | **806,26 €** | |

## Materialkosten

Bei den Materialkosten wird zwischen Kosten für Hardware und Aufwendungen für Software und Lizenzen unterschieden. Hierbei ist zu beachten, dass nur die fertige Webseite mit Serverprogramm ausgeliefert wird. Das Hosting der Webseite bleibt dem Pfadfinderstamm DPSG St. Meinolf überlassen und wird in den Kosten nicht berücksichtigt.

## Hardwarekosten

Die bei dem Projekt verwendete Hardware wird für die Kalkulation linear über 5 Jahre abgeschrieben. Somit ergibt sich folgende Rechnung:

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Anschaffungskosten/Stk.** | **Projektkosten** |
| 2 x MacBook Pro | 1199,00 € | 2,05 € |
|  |  |  |
|  | **Hardwarekosten gesamt:** | **2,05 €** |

## Softwarekosten

Die Softwarekosten setzen sich aus den Lizenzen für Betriebssystem und Anwendungen zusammen. Da das Betriebssystem OS X, welches wir zur Entwicklung verwendet haben, kostenlos von Apple beim Kauf eines MacBooks vertrieben wird, haben wir die Lizenzkosten mit 0,00€ berechnet, da wir die anfallenden Kosten in den Hardwarekosten für die MacBooks zuvor verrechnet haben.

Die lineare Abschreibung beträgt für die Kalkulation auch bei Softwareprodukten fünf Jahre. Daraus ergibt sich folgende Formel zur Berechnung der Softwarekosten des Projekts:

|  |  |  |
| --- | --- | --- |
| **Software** | **Anschaffungskosten/Stk.** | **Projektkosten** |
| 2 x OS X | 0,00 € | 0,00 € |
| 2 x Google Chrome | 0,00 € | 0,00 € |
| NodeJS | 0,00 € | 0,00 € |
| AngularJS 2 | 0,00 € | 0,00 € |
| 2 x Microsoft Word 2013 | 169,00 € | 0,29 € |
| 2 x Microsoft Visual Code | 0,00 € | 0,00 € |
| MySQL | 0,00 € | 0,00 € |
|  | **Softwarekosten gesamt:** | **0,29 €** |

## Sonstige Kosten

Um die Gesamtkosten betrachten zu können, benötigen wir neben den Personal- und den Materialkosten auch weitere Kosten, wie zum Beispiel Miete und Energiekosten.

|  |  |  |  |
| --- | --- | --- | --- |
| **Kostenpunkt** | **Preis** | **Einheit** | **Projektkosten** |
| Miete der Raumfläche | 18,67 € | pro Monat und Mitarbeiter | 8,17 € |
| Reinigung | 5,67 € | pro Monat und Mitarbeiter | 2,48 € |
| Energiekosten | 21,97 € | pro Tag und Büro | 12,82 € |
| Möbel | 0,04 € | pro Tag und Mitarbeiter | 0,24 € |
| Telefon- und Internetkosten | 0,95 € | pro Tag und Mitarbeiter | 16,63 € |
| **Büromaterial** |  |  |  |
| Papier | 3,20 € | pro 500 Blatt | 0,64 € |
| Toner | 137,99 € | 1 Stk. Pro 6000 Blatt | 2,30 € |
| Stifte | 0,89 € | pro Stk. | 3,56 € |
|  |  | **Gesamtkosten:** | **46,84 €** |

## Gesamtkosten

Unter Berücksichtigung aller zuvor genannten Kostenpunkten, berechnen sich Gesamtkosten für das Projekt von insgesamt: **855,44€.**

Da das Projekt für einen ehrenamtlichen Pfadfinderverein erstellt wird, wird nur eine Kostendeckung angestrebt und keine Gewinnmarge auf dem Preis addiert.

# Durchführung und Protokoll des Projektverlaufs

## Ist-Zustands-Analyse

|  |  |
| --- | --- |
| Datum: | Dienstag, 02. Februar 2016 |
| Tätigkeiten: | Analysieren des Ist-Zustandes, Erkennen von Problemen |
| Dauer: | 2,0 Stunden |

Zu Beginn des Projektes haben wir uns zunächst mit unserem Auftraggeber Herr Stöber zusammengesetzt um mit ihm über den aktuellen Zustand der Webseite zu reflektieren. Dabei hatte Herr Stöber bereits einige Probleme aufgedeckt.   
Wir legten ihm während unseres Gesprächs weitere Probleme, wie die nicht vorhandene Umsetzung für mobile Endgeräte, sowie die unübersichtliche Menüstruktur und das veraltete Layout, dar.

Während des gesamten Gespräches machten wir uns Notizen, sodass wir nach dem Gespräch eine Liste mit vorhandenen Problemen hatten, aus welchen wir später eine Anforderungsliste erstellt haben.

## Konzeption des späteren Seitenlayouts

|  |  |
| --- | --- |
| Datum: | Dienstag, 02. Februar 2016 |
| Tätigkeiten: | Konzepte erstellen und diskutieren, Prototyp erstellen |
| Dauer: | 10,0 Stunden |

Nachdem wir im Gespräch die Probleme der aktuellen Webseite festgestellt haben, beginnen wir damit, uns Lösungen für die einzelnen Teilprobleme zu entwickeln.

Das Problem der nicht vorhandenen mobilen Ansicht wollen wir durch den Design-Prozess des mobile-first Designs angehen. Dabei wird die Seite zunächst für ein mobiles Endgerät designt, und wird im folgenden Schritt auch für Desktop-PCs optimiert. Um diese Layouts optimal implementieren zu können, ziehen wir die Nutzung eines Frameworks wie Bootstrap in Betracht.  
Die unübersichtliche Navigation wollen wir durch Dropdown-Menüs ersetzten, um somit klare Hierarchien zu schaffen, und die Navigation an einem zentralen Ort zu haben.  
Um die Inhalte auf der Seite aktueller zu halten und so der Seite neues Leben einzuhauchen, haben wir uns dazu entschieden, dass es vielen Benutzern möglich sein soll, Beiträge zu verfassen und kommentieren. Aus dieser Idee heraus wurde auch die Idee für eine Benutzerverwaltung geboren.

Designtechnisch entscheiden wir uns die Logofarben der DPSG (dunkelblau & bordeauxrot) aufzugreifen und zu benutzen. Da die Seite zu einem Jugendverband gehört, entscheiden wir uns zudem zum Einsatz von großen Fotos im Header und weiteren Seiten.

## Erstellung des Pflichtenhefts

|  |  |
| --- | --- |
| Datum: | Dienstag, 02. Februar 2016 - Mittwoch, 03. Februar 2016 |
| Tätigkeiten: | Lösungsansätze in konkrete Anforderungen umsetzen, Testfälle erstellen, Techniken festlegen |
| Dauer: | 10,0 Stunden |

Als wir mit unserem Konzept für die zukünftige Webseite fertig und den ersten Prototypen fertiggestellt haben, haben wir uns um die Erstellung des Pflichtenhefts gekümmert.   
Hierzu haben wir die in der Konzeptphase besprochenen Lösungsansätze in konkrete Anforderungen an die Webseite überführt.

Nachdem wir mit der Überführung der Anforderungen fertig waren, haben wir diese nach Musskriterien, Wunschkriterien und Abgrenzungskriterien aufgeteilt und sie somit priorisiert.  
Die Musskriterien beinhalten dabei die Funktionen, die wir auf jeden Fall in unserem Projekt umsetzen müssen. Dies steht im Gegensatz zu den Wunschkriterien, die nicht essentielle Features beschreiben. In den Abgrenzungskriterien werden Kriterien benannt, die nicht möglich sein sollen bzw. können (zum Beispiel sollen unangemeldete Benutzer keine Newseinträge löschen können).

Zudem haben wir uns auch Gedanken gemacht, wo die Webseite während und nach der Projektdurchführung gehostet werden soll und welche Zielgruppen wir primär ansprechen wollen.

Anschließend haben wir die Produktumgebung (Software, Hardware, Produktschnittstellen) der Webseite beschrieben und die zuvor erstellten Muss-, Wunsch- und Abgrenzungskriterien in detaillierte Funktionen aufgeteilt und aufgeschrieben. Um ein qualitativ hochwertiges Produkt, welches möglichst wenig Fehler aufweist, dem Kunden bieten zu können, haben wir für jede zuvor definierte Funktion einen oder mehrere Testfälle beschrieben.

Nach Fertigstellung des Pflichtenhefts haben wir dieses zur Kontrolle an Herr Schlichting weitergegeben und um seine Meinung gebeten. Da Herr Schlichting nur kleinere Verbesserungsvorschläge hatte, haben wir diese in das Pflichtenheft eingebracht und anschließend an Herr Stöber weitergegeben. Dieser war mit dem Pflichtenheft äußerst zufrieden, somit konnten wir mit der Umsetzung der Webseite beginnen.

## Datenbankentwurf

|  |  |
| --- | --- |
| Datum: | Donnerstag, 04. Februar 2016 - Dienstag, 09. Februar 2016 |
| Tätigkeiten: | Modellierung ER-Diagramm, Erstellung der Datenbank in MySQL |
| Dauer: | 5,0 Stunden |

Um die Daten für unsere Webseite zu speichern und sinnvoll zu verknüpfen haben wir uns dazu entschieden eine SQL-basierte Datenbank zu wählen. Da MySQL kostenlos verfügbar ist und eine große Community mit vielen Tools hat, haben wir uns entschieden MySQL zu verwenden.  
Anschließend sind wir durch die vorher festgelegten Funktionen gegangen und haben Entitätstypen festgelegt, die notwendig sind um die Daten, die für die Funktionen benötigt sind, speichern zu können. Diesen ersten Entwurf haben wir dann in die 3. Normalform gebracht.

Der fertige Datenbankentwurf wurde dann noch dem Pflichtenheft zugefügt und beschrieben.  
Anschließend haben wir die Datenbank in MySQL erstellt und von der fertig angelegten Datenbank ein Dumpfile erstellt, das die Entitäten der Datenbank darstellt und in installierte MySQL-Server importiert werden kann, um die Datenbank auf diesem Server anzulegen. Dieses Dumpfile haben wir in GitHub eingecheckt und so die Datenbank portabel gemacht.

## Implementierung

|  |  |
| --- | --- |
| Datum: | Dienstag, 09. Februar 2016 - Dienstag, 16. Februar 2016 |
| Tätigkeiten: | Erstellen des Designs, Implementierung des Servers, Datenbankabfragen, Authentifizierung, Umsetzung aller Musskriterien |
| Dauer: | 50,0 Stunden |

Die Implementierung unserer Webseite haben wir damit begonnen, den NodeJS-Server aufzusetzen. Diesen haben wir mit Hilfe von ExpressJS zu einem Webserver gemacht und zunächst jede Anfrage mit einer statischen HTML-Datei beantwortet. Wenn der Server dann gestartet wurde, konnte die Seite über localhost:3000 angesehen werden, der erste Schritt war geschafft.

Der nächste Schritt war in der gesendeten Datei eine Angular Applikation zu starten, damit diese im Client laufen kann. Der Vorteil von Angular Applikationen ist, dass sie alle Daten im Hintergrund laden und daher nur Templates und JavaScript nachgeladen werden müssen, und nicht die komplette Seite neugeladen werden muss. Nachdem die grundsätzliche Applikation implementiert war, musste die Angular Applikation weitere Template-Dateien und JavaScript-Dateien vom Server laden können. Dazu haben wir einen Ordner auf dem Dateisystem des Servers statisch auf eine bestimmte URL gebunden, sodass alle Dateien innerhalb dieses Ordners über den Webserver geladen werden können.

Unser nächstes Ziel war es, Navigation in unsere Webseite zu bringen. Dabei war es wichtig, dass einerseits über die eingebaute Navigation navigiert werden konnte, jedoch auch, dass die URL zu jedem Zeitpunkt korrekt war, sodass Links versendet werden können und man wieder auf der gleichen Seite herauskommt. Wir haben die Navigation komplett in Angular umgesetzt, welches einen grundlegenden Rahmen lädt und dann die Templates und JavaScript-Dateien der jeweiligen Seite nachlädt. Dadurch funktioniert die Navigation sehr schnell und fast ohne Ladezeiten.

Nachdem die Navigation funktionsfähig war, war es unser nächstes Ziel, Daten aus der Datenbank zu laden und auf der Webseite anzeigen zu können. Dazu haben wir arbeitsteilig gearbeitet. Wir haben zunächst auf dem NodeJS-Server eine API-Schnittstelle definiert, die von der Angular Applikation abgefragt werden kann. Der eine hat dann die Logik zur Abfrage aus der Datenbank und zum Versenden der Daten implementiert, der andere hat in der Angular Applikation die Daten von der Schnittstelle abgefragt und diese dann entsprechend angezeigt. Dieses haben wir zunächst mit der Schnittstelle für die Newsartikel implementiert, weiter API Schnittstellen folgten dann für Kommentare und Gruppenseiten.

Eine besondere Herausforderung war das Login-System, da wir dieses Session basiert gestalten wollten. Unsere Lösung dafür war, nach erfolgreichem Login einen verschlüsselten Cookie im Browser zu speichern, der Informationen wie Nutzername und Rolle des eingeloggten Users speicherte. Der Server kann dann validieren, ob dieser Sessioncookie vorhanden ist, und entsprechend bestimmte API-Abfragen erlauben oder verbieten.  
Die Sessioncookies verarbeiten wir mit Hilfe eines NodeJS-Moduls, die Cookies laufen momentan nicht aus, dies könnte jedoch einfach umkonfiguriert werden.  
Auf Clientseite nutzen wir den in HTML5 integrierten Sessionspeicher localStorage um Benutzerinformation zu speichern und in Angular zu überprüfen wie bestimmte Seiten angezeigt werden müssen. Dieses System ist zwar nicht vollständig sicher auf Clientseite, bringt jedoch nicht viel, da der Server Anfragen erneut validiert und dann nicht zulässt.

## Testphase

|  |  |
| --- | --- |
| Datum: | Dienstag, 23. Februar 2016 |
| Tätigkeiten: | Testen der implementierten Funktionen |
| Dauer: | 5,0 Stunden |

Nachdem wir die Implementierung der zuvor definierten Musskriterien erfolgreich fertiggestellt haben, haben wir die von uns erstellte Webseite mithilfe von manuellen Tests ausgiebig getestet.  
Wir sind dabei die im Pflichtenheft beschriebenen Testfälle durchgegangen.

Zunächst haben wir versucht einen neuen Benutzer auf der Webseite mit einem bereits vorhandenen Nutzernamen zu registrieren, was richtigerweise auch fehlschlug. Auch, wenn man alle Felder bei der Registrierung leer gelassen hat, war es nicht möglich sich zu registrieren.   
Erst nachdem wir gültige Werte in die Eingabefelder eingetragen hatten, war die Registrierung erfolgreich und wir waren eingeloggt.

Danach haben wir die Newsfunktion ausgiebig getestet und sind zu dem Ergebnis gekommen, dass alle Testfälle erfolgreich durchlaufen wurden. Wir konnten nur neue Newseinträge erstellen, wenn alle Felder ausgefüllt waren. Zudem konnten wir nur unsere eigens erstellten Newseinträge bearbeiten und löschen. Von anderen Nutzern verfasste Newseinträge konnten von uns nicht verändert werden.

Bei der Kommentarfunktion verlief es ähnlich wie mit der Newsfunktion. Eigene Kommentare erstellen konnten wir unter jedem Newsbeitrag, wenn alle Felder zuvor ausgefüllt wurden. Fremde Kommentare konnten gelesen, jedoch im Gegensatz zu eigens erstellten, nicht bearbeitet oder gelöscht werden.

Da unser zuvor neu angelegter Benutzer standardmäßig die Roller „User“ besitzt, konnten wir zunächst nicht die Gruppenseiten bearbeiten, was ein gewolltes Verhalten ist. Daher mussten wir unserem Benutzer die Rolle „Group-Admin“ in der MySQL Datenbank zuweisen. Nachdem dieser die neue Rolle erhalten hatte, konnten wir die Gruppenseiten erfolgreich bearbeiten.

Nachdem wir mit unseren zuvor definierten Testfällen aus dem Pflichtenheft fertig waren, haben wir uns von der Webseite ausgeloggt und versucht neue Newseinträge und Kommentare als nicht angemeldeter Nutzer zu verfassen. Dies ist uns, genau so wie das löschen und bearbeiten fremder Beiträge, nicht gelungen.

Daher sind wir zu dem Ergebnis gekommen, dass die von uns implementierten Funktionen, korrekt ausgeführt und Falscheingaben vom Server abgefangen werden.

## Erstellung der Projektdokumentation

|  |  |
| --- | --- |
| Datum: | Dienstag, 23. Februar 2016 – Dienstag, 1.März 2016 |
| Tätigkeiten: | Erstellen der Projektdokumentation, Zusammenfassen der Ergebnisse, Erstellen der Benutzerdokumentation |
| Dauer: | 18,0 Stunden |

Zum Schluss unseres Projekts, haben wir unsere Projektdokumentation anhand zuvor gemachter Notizen angefertigt. In unserer Projektdokumentation haben wir alle Tätigkeiten innerhalb der einzelnen Projektphasen detailliert beschrieben und erläutert.

Zudem haben wir die Ausgangslage, die Projektziele und die für das Projekt benötigte Hardware, Software und weitere Hilfsmittel beschrieben. Des Weiteren haben wir im Pflichtenheft unseren zu Beginn angefertigten Projektplan mit den tatsächlich angefallenen Stunden verglichen und erläutert, wieso wir an einigen Projektphasen zunächst falsch kalkuliert und entweder mehr oder weniger Zeit für die jeweilige Phase benötigt haben.

Anschließend haben wir die Benutzerdokumentation erstellt. Die Zielgruppe dieser Benutzerdokumentation sind alle Personen, die die Seite benutzen wollen. Da diese Zielgruppe nicht näher auf einen bestimmten Personenkreis eingegrenzt werden kann, gehen wir davon aus, dass der Leser dieser Dokumentation wenig bis gar kein technisches Verständnis besitzt. Daher verwenden wir in der Benutzerdokumentation möglichst viele Bilder und gehen nicht auf technische Begebenheiten ein.

Nach der Fertigstellung der Benutzerdokumentation, haben wir diese in den Anhang der Projektdokumentation eingefügt.