1 Thematisierung und Projektumfeld

1.1 Thematisierung

1.2 Projektumfeld

1.2.1 Projektteam

1.2.2 Auftraggeber

1.2.3 Softwareumgebung

2 Planung

2.1 Zeitplanung

2.2 Ressourcen

2.3 Kosten

3 Lösungskonzept

3.1 Ziele

3.1.1 Fehlerfreie Software

3.1.2 Barrierefrei

3.1.3 Usability

3.1.4 Datenschutz

3.2 Funktionalitäten

3.3 Diagramme

3.4 Modelle

4 Realisierung

4.1 Gestaltung der grafischen Oberfläche

4.2 Umsetzung

4.3 Probleme

5 Tests

5.1 Testkonzepte

5.2 Testfälle

5.3 Testprotokolle

6 Projektergebnis

6.1 Soll/Ist Vergleich

6.2 Verlauf

6.3 Ergebnis

6.4 Fazit

7 Eidesstattliche Erklärung

8 Anhänge

8.1 Lastenheft

8.2 Pflichtenheft

8.3 Angebot

8.4 Installationsanleitung

8.5 Bedienungsanleitung (Deutsch)

8.6 Bedienungsanleitung (Englisch)

8.7 Testprotokolle

8.8 Teamprotokolle

1 Thematisierung und Projektumfeld

1.1 Thematisierung

1.2 Projektumfeld (Stakeholdererwartungen, Projektrisiken)

1.2.1 Projektteam

|  |  |  |
| --- | --- | --- |
| Name | Rolle | Firma |
| Christian Ehringfeld | Back-End | Bundeskriminalamt |
| David Mock | Front-End | Bundesamt für Wirtschaft und Ausfuhrkontrolle |
| Matthias Unterbusch | Projektmanager | Hilscher Gesellschaft für Systemautomation mbH |

1.2.2 Auftraggeber

Der Auftraggeber und die Brühlwiesenschule in Hofheim. Dabei ist der Ansprechpartner Herr Wild. Der Betreuer des Projekts ist Herr Grüning.

1.2.3 Softwareumgebung

Da die Software als Webapplikation realisiert werden sollte, fiel die Wahl der Sprache auf PHP (Back-End) und HTML (Front-End). Um einige zusätzliche Funktionen auf der Webseite bereitzustellen, kommt außerdem JavaScript zum Einsatz. Außerdem wurde ein Apache Webserver benutzt.

2 Planung

2.1 Zeitplanung

GANTT Diagramm?

Tabelle mit Meilensteinen

Wann wird getestet?

2.2 Ressourcen

Als Ressourcen für das Projekt werden nur die Sachmittel benötigt. Diese umfassen einen normalen Bildschirmarbeitsplatz. Dazu gehören Tische, Stühle und PCs, die komplett von der Schule zur Verfügung gestellt wurden. Zudem wurde eine Verbindung mit ins Internet mit Hilfe eines WLAN-Hotspots zur Verfügung gestellt.

Da das Projekt auch einige Arbeitszeit zu Hause erforderte und die PCs der Schule durch eine starke Netzauslastung sehr langsam waren, haben sich die Mitarbeiter darauf verständigt, dass mit privaten Notebooks gearbeitet wird.

Außerdem wurde für Tests ein Webserver und eine Versionsverwaltung verwendet. Dazu wurde das webbasierte Projektmanagementtool Redmine benutzt, welches schon von einem Mitglied des Projektteams konfiguriert wurde.

2.3 Kosten

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Pos** | **Anzahl** | **Einheit** | **Beschreibung** | **Netto** |
| 1 | 40 | Stunden | Erstellung des Pflichtenhefts | 2400 |
| 2 | 8 | Stunden | Erstellung eines Angebots | 480 |
| 3 | 16 | Stunden | Benutzerhandbuch (mehrsprachig) | 960 |
| 4 | 8 | Stunden | Erstellen einer Installationsanweisung | 480 |
| 4 | 32 | Stunden | Erstellung einer Dokumentation | 1920 |
| 5 | 8 | Stunden | Datenbankmodellierung | 480 |
| 6 | 16 | Stunden | Benutzerverwaltung | 960 |
| 7 | 32 | Stunden | Terminverwaltung | 1920 |
| 8 | 16 | Stunden | Administrationsbereich | 960 |
| 9 | 8 | Stunden | Datenimport | 480 |
| 10 | 16 | Stunden | Rechteverwaltung | 960 |
| 11 | 24 | Stunden | Erstellen von mock ups | 1440 |
| 12 | 24 | Stunden | Erstellen von Prototypen | 1440 |
| 13 | 16 | Stunden | Lehrerfiltererstellung | 960 |
| 14 | 8 | Stunden | Formelementanpassung | 480 |
| 15 | 32 | Stunden | Integration von Prototypen | 1920 |
| 16 | 48 | Stunden | Systemtest | 2880 |
|  | **352** | **Stunden** |  |  |
| 2% sKonto bei Zahlung innerhalb von 14 Tagen | | | | |
| **Gesamt Netto** | | |  | **21120,00** |
| **19 % Umsatzsteuer** | | |  | **4012,80** |
| **Gesamt Brutto** | | |  | **25132,80** |

3 Lösungskonzept

3.1 Ziele

Ziel ist es eine Elternsprechtagssoftware zu entwickeln, um die Handhabung des Elternsprechtags zu vereinfachen. Die Elternsprechtagssoftware soll es ermöglichen online Termine für den Elternsprechtag zu vereinbaren. Dazu wird eine Webseite entworfen, für die lediglich ein Internetzugang nötig ist. Dabei wird berücksichtigt, dass die Zielgruppe in der Regel Berufsfremd ist.

3.1.1 Fehlerfreie Software

Ein Hauptziel ist eine fehlerfreie Software. Da der Elternsprechtag ein offizieller Anlass ist und die Schule nach außen hin repräsentiert, dürfen an diesem Tag natürlich keine Unstimmigkeiten auftauchen, die auf die Elternsprechtagssoftware zurückzuführen sind.

3.1.2 Barrierefreiheit

Ein weiteres wichtiges Ziel ist die Barrierefreiheit. Dabei ist es sehr wichtig, dass jeder, auch Menschen mit Behinderungen, die Webseite in vollem Umfang nutzen können. Dabei kann vor allem mit einem einfachen Design, mit möglichst starken Kontrasten darauf geachtet werden, dass auch Menschen mit einen Sehschwäche die Website in vollem Umfang benutzen können.

Ein weiterer wichtiger Punkt der Barrierefreiheit ist die Verwendung von JavaScript. JavaScript muss zusätzlich auf jedem System installiert werden und erhöht somit die Anforderungen an die Benutzer. Daher funktioniert die gesamte Webseite auch ohne JavaScript. JavaScript stellt dabei für die Benutzer lediglich einige zusätzliche Funktionen zur Verfügung. Ein Beispiel dafür ist das zusätzlich einblendbare Menü, welches ein besseres Design hat. Da es nicht immer eingeblendet ist, steht der zusätzliche Platz für den Rest der Website zur Verfügung.

Ein weiteres Feature, welches nur mit JavaScript genutzt werden kann, gibt es bei der Termineintragung. Mit aktivierten JavaScript können freie Termine angeklickt werden, worauf die Felder Datum und Uhrzeit automatisch ergänzt werden. Wenn kein JavaScript aktiviert ist müssen diese Felder mit Hilfe von Drop Menüs manuell ausgefüllt werden.

3.1.3 Usability

Die Benutzerfreundlichkeit soll auf jeden Fall gewährleistet sein. Dazu trägt vor allem ein schlichtes Design der Seite bei, wobei vor allem auf klare Kontraste geachtet wurde. Außerdem wird eine sehr einfache Menüstruktur geachtet. Beim ersten Login werden die Eltern direkt aufgefordert ein Kind einzugeben und es muss nicht erst im Profil mühselig nachgetragen werden. Des Weiteren werden, falls Termine gemacht wurden, diese direkt im Startbildschirm angezeigt und es muss nicht jeder Lehrer bei dem Termine gemacht wurden, einzeln überprüft werden.

3.1.4 Datenschutz

Ein sehr wichtiger Punkt ist der Datenschutz der Schüler, Eltern und Lehrer. Deshalb werden die Zeitpläne der Lehrer den Eltern nicht komplett angezeigt, um nicht jedem alle Informationen zugänglich zu machen. Die belegten Termine werden nur als belegt gekennzeichnet und nicht genau angezeigt wer welchen Termin belegt hat.

Außerdem werden die Passwörter in der Datenbank verschlüsselt gespeichert. Dabei werden gesalzene sha512 Hashes verwendet.

Da die Daten maskiert werden, ist eine SQL-Injection nicht möglich. Da alle Ausgaben maskiert werden ist Cross Site Scripting ebenfalls nicht möglich.

3.2 Funktionalitäten

3.3 Diagramme

Klassendiagramme

Anwendungsfalldiagramm

Aktivitätsdiagramm

3.4 Modelle

Datenbankmodell

4 Realisierung

4.1 Gestaltung der grafischen Oberfläche

Bei der Gestaltung der grafischen Oberfläche wurde vor allem auf ein einfaches und schlichtes Design geachtet. Die Eltern sollen nicht durch ein zu aufdringliches Design verunsichert werden. Dabei wurde vor allem auf klare Kontraste viel Wert gelegt. Es wurde nur ein schlichtes Logo für die Software entworfen, welches sich eher unauffällig auf der Webseite präsentiert. Da es sich um eine offizielle Schulhomepage handelt, wurde ebenfalls das Logo der Brühlwiesenschule auf der Webseite integriert, welches sich ähnlich wie das Logo der Elternsprechtagssoftware im Hintergrund hält.

Um die Benutzerfreundlichkeit weiter zu verbessern, gibt es auf vielen Seiten eine Box, in welcher eine kurze Beschreibung vorhanden ist, welche Aktionen auf dieser möglich sind und wie diese auszuführen sind. Weiterhin wurde zu jedem Feld eine Beschreibung hinzugefügt, die dem Benutzer angibt, welche Information in dieses Feld hingeschrieben werden muss. Zudem wurden die Seiten nicht mit Funktionen über und relativ simpel gehalten.

4.2 Umsetzung

Die Umsetzung der Software wurde anhand der Muss-, Soll-, und Kannkriterien realisiert. Der erste Schritt war die Modellierung einer Datenbank, um alle anderen Funktionen zu ermöglichen. Danach wurde die Möglichkeit geschaffen sich zu registrieren, um einen internen Bereich zu schaffen, sodass jeder Benutzer einen Bereich für sich hat. Damit wurde die Grundlage für den Datenschutz geschaffen. Zeitgleich wurde das Design für die Seite fertiggestellt und mit der Erstellung der Prototypen begonnen. Anschließend wurde mit der Rechteverwaltung begonnen, um mehrere Rollen zu realisieren.

Anschließend wurde das Eintragen von Terminen umgesetzt. Dazu wurde ein Lehrerfilter erstellt, um einen Lehrer auszuwählen. Danach wurden die Zeitpläne für Lehrer erstellt und das Termineintragen von der Rolle Eltern möglich gemacht. Währenddessen wurden die Prototypen integriert und Formelemente angepasst. Zum Schluss wurde der Datenimport in die Software integriert. Die Accounts der Lehrer können automatisch aus einer .csv Datei erstellt werden.

4.3 Probleme

Ein Problem entsteht mit der Kodierung in der Datenbank. Zunächst wurde UTF8-General-CI benutzt. Dieser Zeichensatz ist case-insensitive und daher gut für eine Datenbank geeignet. Außerdem ist diese Kodierung schneller, da nicht auf Bitebene verglichen wird. Das führte zu Problemen bei der Auswahl der Lehrer, die mit Umlauten beginnen von denen zu unterscheiden, die mit dem entsprechendem Vokal beginnen, da UTF8-General-CI Umlaute als normale Vokale behandelt, z.B. Ä wie A. Aus diesem Grund wurde der Zeichensatz auf UTF8 binär gewählt, bei dem auf Bitebene verglichen wird, und somit Umlaute korrekt verarbeiten kann.

Ein weiteres Problem waren Elternsprechtage mit unterschiedlicher Termindauer.

Mehrere Elternsprechtage, unterschiedliche Terminzeiten im View tauchen Probleme auf

Rechteverwaltung, falsche Anwendungsstruckter, zu überladen

5 Tests

5.1 Testkonzepte

Während der Programmierung wurden einzelne Teile des Programms immer wieder einem Modultest unterzogen. Dieser wurde von den Entwicklern mit Hilfe des Whiteboxtestverfahrens durchgeführt. Außerdem wurde bei der Zusammenführung des Front-Ends und Back-Ends ein Integrationstest von den Programmierern durchgeführt.

Um die Software so fehlerfrei wie möglich zu gestalten, wurde zum Schluss noch ein Systemtest durchgeführt. Dieser wurde als Blackboxtest durchgeführt und vom Projektmanager durchgeführt.

**5.2 Testfälle**

5.2.1 Administrationsbereich

Als erstes wird der Administrationsbereich getestet. Das geschieht deshalb, dass zunächst ein Elternsprechtag erstellt und eine Lehrerliste eingelesen werden muss, damit die Software wie geplant funktionieren kann.

Dazu gehört als erstes das Erstellen eines Elternsprechtages. Beim Eintragen des Datums und der Zeiten ist ein sehr spezielles Format erforderlich. Aus diesem Grund werden mehrere falsche Eingaben getätigt, um zu überprüfen, ob ein falsches Datum festgelegt werden kann. Zum Schluss wird ein korrektes Datum eingegeben, um zu testen, ob ein Elternsprechtag korrekt angelegt wird.

Danach wird das Einlesen der Lehrerliste getestet. Dabei wird zunächst versucht eine falsche Datei auszuwählen. Anschließend wird eine korrekte Datei eingelesen und geprüft, ob die Lehrerdaten korrekt eingelesen werden

5.2.2 Rolle Eltern

Bei den Eltern wird zunächst das Eintragen von Kindern überprüft. Da an dieser Stelle keine Überprüfung stattfindet, ob die Kinder in der Datenbank vorhanden sind, können an dieser Stelle keine „falschen“ Eingaben gemacht werden. An dieser Stelle muss nur geprüft werden, wie viele Kinder maximal eingetragen werden können.

Anschließend wird die Verwaltung des eigenen Accounts getestet. Dazu wird das Menü „Ihr Account“ aufgerufen und getestet, welche Daten verändert werden können. Danach werden mehrere Daten geändert und gespeichert.

Als letztes wird die Termineintragung getestet. Dabei wird geprüft, ob der Lehrerfilter korrekt funktioniert oder Fehler auftreten. Danach wird ein Lehrer ausgewählt und versucht einen bereits belegten Termin einzutragen. Zum Schluss wird ein freier Termin reserviert und auf die Seite „Ihre Termine“ gewechselt, um zu überprüfen, ob der Termin korrekt eingetragen wurde.

5.2.3 Rolle Lehrer

Bei den Lehrern wird ebenfalls die Accountverwaltung des eigenen Accounts. Dabei wird wieder getestet, welche Daten verändert werden können. Dann werden wieder mehrere Daten geändert, gespeichert und geprüft, ob alles korrekt übernommen wurde.

Des Weiteren wird überprüft, ob der Zeitplan des Lehrers korrekt angezeigt wird und ausgedruckt werden kann.

Zum Schluss wird die Tanverwaltung getestet. Als Testeingaben werden dabei 0, eine Zahl über 100 und letztendlich eine Zahl zwischen 1 und 100.

5.2.4 Verwaltung

Bei der Verwaltung wird als erstes die Benutzerverwaltung getestet. Dabei wird die detaillierte Benutzeransicht zuerst getestet. Danach wird das Bearbeiten eines Benutzers getestet. Dabei wird vor allem das Ändern der E-Mail Adresse geprüft, da die anderen Daten schon bei den Rollen Eltern und Lehrer genau geprüft wurden. Dann wird überprüft, ob ein Benutzer gelöscht werden kann.

Ferner wird die Termineintragung einem Test unterzogen. Dazu wird zunächst versucht ein Termin unter Angabe von inkorrekten Daten anzulegen, um prüfen, wie die Software reagiert. Am Ende werden mehrere Testtermine mit korrekten Eingaben erstellt und überprüft, ob die Termine korrekt übernommen wurden.

5.3 Testprotokolle

Der abschließende Systemtest wurde genutzt, um eventuelle Fehler aus der Software zu entfernen und um die Benutzerfreundlichkeit zu erhöhen.

Die Testprotokolle sind im Anhang unter 8.7 vorhanden.

6 Projektergebnis

6.1 Soll/Ist Vergleich

Zeiten

6.2 Verlauf

6.3 Ergebnis

6.4 Fazit

7 Eidesstattliche Erklärung

8 Anhänge

8.1 Lastenheft

8.2 Pflichtenheft

8.3 Angebot

8.4 Installationsanleitung

8.5 Bedienungsanleitung (Deutsch)

8.6 Bedienungsanleitung (Englisch)

8.7 Testprotokolle

8.8 Teamprotokolle