|  |  |  |
| --- | --- | --- |
| Machbarkeit | 12. Januar  2015 | |
| [Geben Sie hier das Exposee für das Dokument ein. Das Exposee ist meist eine Kurzbeschreibung des Dokumentinhalts. Geben Sie hier das Exposee für das Dokument ein. Das Exposee ist meist eine Kurzbeschreibung des Dokumentinhalts.] | | Jevtic, Kodras, El-FAR & Stokic |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Version* | *Autor* | *Datum* | *Status* | *Kommentar* |
| 1.0  1.1  1.2 | B. Jevtic  B. Jevtic | 08.02. 2014  08.03.2014 | Beispiel  Progress | Grobe Anfänge |

Inhaltsverzeichnis

[1.1 Projekt-Team 4](#_Toc408842619)

[1.2 Projektbeschreibung 5](#_Toc408842620)

[2.1 Ist-Erhebung 5](#_Toc408842621)

[2.1.1 Allgemeine Ist-Erhebung 5](#_Toc408842622)

[2.1.2 Ist-Zustand 6](#_Toc408842623)

[2.2 Soll-Zustand 6](#_Toc408842624)

[2.2.1 Allgemeine Beschreibung 6](#_Toc408842625)

[2.2.2 Muss-Ziele 7](#_Toc408842626)

[2.2.3 Kann-Ziele 8](#_Toc408842627)

[2.2.4 Nicht-Ziele 8](#_Toc408842628)

[2.3 Festlegen der Hauptdaten 8](#_Toc408842629)

[2.4 Festlegen der Hauptleistungen 9](#_Toc408842630)

[2.5 Festlegen der wichtigsten Qualitätsmerkmale 9](#_Toc408842631)

[3.1 Prüfen alternativer Lösungsvorschläge 10](#_Toc408842632)

[3.1.1 Programmiersprachen/Skriptsprachen 10](#_Toc408842633)

[3.1.2 Datenbank Management System (DBMS) 13](#_Toc408842634)

[3.1.3 Gesamtfazit 17](#_Toc408842635)

[3.2 Prüfen der technischen Durchführbarkeit 18](#_Toc408842636)

[3.2.1 Technologien / Softwaretechnische Durchführbarkeit 18](#_Toc408842637)

[3.2.2 Technologien / Hardwaretechnische Durchführbarkeit 19](#_Toc408842638)

[3.2.3 Verfügbarkeit von Entwicklungs- und Zielmaschinen 19](#_Toc408842639)

[3.3 Prüfen der personellen Durchführbarkeit 20](#_Toc408842640)

[3.3.1 Qualifikation der Fachkräfte 20](#_Toc408842641)

[3.3.2 Zusammenarbeit der Teammitglieder 20](#_Toc408842642)

[3.4 Prüfen der Risiken 21](#_Toc408842643)

[3.4.1 Personelle Risiken 21](#_Toc408842644)

[3.4.2 Technische Risiken 21](#_Toc408842645)

[3.5 Prüfen der ökonomischen Durchführbarkeit 23](#_Toc408842646)

[3.5.1 Terminschätzung 23](#_Toc408842647)

[4.1 Nutzen für den Kunden 26](#_Toc408842648)

1. Projektdaten

# 1.1 Projekt-Team

**Name:** Boban Jevtic

**E-Mail:** [bjevtic@student.tgm.ac.at](mailto:bjevtic@student.tgm.ac.at)

**Fähigkeiten:** Java, Photoshop, CSS, PHP, MySQL, HTML, JavaScript, C

**Bisherige Projekte:** Subnet-Rechner

**Name:** Dominik Kodras

**E-Mail:** [dkodras@student.tgm.ac.at](mailto:dkodras@student.tgm.ac.at)

**Fähigkeiten:** Java, Photoshop, CSS, PHP, MySQL, HTML, JavaScript, C

**Bisherige Projekte:** JPS-Wecker, The Lightbringer

**Name:** Stefan Stokic(Programmierer)

**E-Mail:** [sstokic@student.tgm.ac.at](mailto:sstokic@student.tgm.ac.at)

**Fähigkeiten:** Java, Photoshop, CSS, PHP, MySQL, HTML, JavaScript, C

**Bisherige Projekte:** Space Warfare, FlashNotify+

**Name:** Matthias El-FAR

**E-Mail:** [melfar@student.tgm.ac.at](mailto:melfar@student.tgm.ac.at)

**Fähigkeiten:** Java, Photoshop, CSS, PHP, MySQL, HTML, JavaScript, C

**Bisherige Projekte:** Vokabeltrainer, NotiZync

# 1.2 Projektbeschreibung

3D-Schach ist ein dreidimensionales Brettspiel, das auf mehreren Ebenen basiert. Es besteht aus 32 Spielfiguren (16 weiße Figuren und 16 schwarze Figuren). Es gibt 6 Arten von Spielfiguren (Bauer, Turm, Pferd, Läufer, Dame und König). Jede Spielfigur hat bestimmte Regel, wie man sie bewegen darf. Die Art des 3D-Schachs heißt Star-Track-Schach. Das ist eine besondere Variante des klassischen Schachspiels, die immer wieder in populären Medien auftaucht. Die Regeln beim 3D-Schach sind ähnlich, wie beim 2D-Schach, werden aber ergänzt, da es mehrere Ebenen gibt.

Außerdem bieten wir den Spieler die Möglichkeit gegen einen anderen Spieler zu spielen, aber auch gegen eine künstliche Intelligenz zu spielen.

Die Software wird später komplett kostenlos online zur Benutzung verfügbar sein.

2. Voruntersuchung des Produkts

# 2.1 Ist-Erhebung

## 2.1.1 Allgemeine Ist-Erhebung

Als Anwendung wird dieses Spiel ein ausführbares JAR-File sein, dass heißt der Benutzer kann dieses Spiel lokal auf dem PC ausführen, jedoch unter der Voraussetzung, dass er eine aktuelle Version von JRE (Java-Laufzeitumgebung) auf seinem PC installiert hat.

## 2.1.2 Ist-Zustand

* Nachdem sich der User registriert hat und er die zugeschickte E-Mail auf seinem E-Mail-Account bestätigt hat, kann er sich seinem Benutzernamen und Passwort einloggen und online gegen andere Spieler antreten.
* Es gibt schon viele Schachspiele die am Computer visualisiert wurden, jedoch gibt es nur wenige andere Arten von Schachspielen, die vom herkömmlichen Schach abweichen.
* Die Idee für ein dreidimensionales Schach ist relativ alt. Es gibt schon sehr viele Schachspiele, die auf dem Markt verkauft werden. Jedoch sind diese vermarkteten 3D-Schach-Spiele ziemlich benachteiligt, weil sie eine Menge Geld kosten.
* Im Internet stehen auch Schach-Spiele in der 3D Umgebung als Download zur Verfügung existieren.
* Im Netz findet man nicht nur Schachspiele, wo man nur gegen eine künstliche Intelligenz spielen kann, sondern man kann sich auch gegen menschliche Gegner messen.
* Als Beispiel nennen wir einen „Star-Trek-Schach“ mit folgenden Eigenschaften
* sehr bekannte 3D-Variante, keine offiziellen Regeln.
* Bezeichnung: Herkunft aus der Fernsehserie „Star-Trek“
* 64 Felder, gleiche Figuren, wie beim 2D-Schach
* Übereinanderliegende Brettern mit vier beweglichen „Angriffsdecks“
* Jede Figur darf über Ebenen ziehen

# 2.2 Soll-Zustand

## 2.2.1 Allgemeine Beschreibung

* Das Ziel des Projektes das Laufen des Spiels unter jedem Betriebssystem. Es soll auch die Möglichkeit haben mit anderen Spielern anzutreten und sich mit Ihnen zu verbinden.
* Jedoch ist die unterscheidet sich schon unser Programm von anderen Spielen, indem dieses 3D-Schach ein OpenSource sein wird.

## 2.2.2 Muss-Ziele

* Benutzerfreundlichkeit

Um die Anwendung benutzerfreundlicher zu machen muss das Spiel plattformunabhängig sein und auf jedem Betriebssystem funktionieren. Um den Gegenspieler erreichen zu können, benötigen wir also einen Web Server, um die Verbindung aufzubauen.

* Ermöglichen eine einfache Registrierung und die Verbindung mit anderen Spieler

Für die Benutzung von dem Spiel wird ein Account benötigt. Dieser Account wird über eine Registrierung erstellt, die am Anfang des Programms angezeigt wird und besitzt Daten über den User wie Username und Passwort.

* Interface für die Kommunikation mit anderen Spieler

Das Programm muss die Möglichkeit besitzen auf irgendeiner Art und Weise mit anderen Menschen (Usern) zu kommunizieren. Dabei wird eine Chat-Funktion in diesem Spiel integriert sein.

* Erweiterung um eine Koordinate (wie zum Beispiel z-Achse)

Ein 3D-Schach-Spiel ist ein dreidimensionales Spiel, das auf mehreren bzw. (mindestens 2 Ebene) basiert. Man kann sich das so vorstellen, wie wenn das Spiel um eine Koordinate erweitert wird.

* Darstellung

Das dreidimensionale Schachspiel, Registrierung und Login muss grafisch dargestellt werden. Nach dem Login sollen alle offenen Spielräume dargestellt werden.

* Validierung der Spielzüge

Die Applikation soll den Spielzug überprüfen, ob der Zug gültig ist oder nicht. Wenn der Spielzug ungültig ist, wird er nicht ausgeführt.

## 2.2.3 Kann-Ziele

* Darstellung des Schachspiels mit mehreren Farben

Dem Benutzer wird die Auswahl von Farben zur Verfügung gestellt, um das Spiel zu personalisieren. Dabei kann er die Farben der Figuren und des Schachbretts verändern.

* Registrierung mit Hilfe von Mail-Verifikation

Bei der Registrierung muss der User zusätzliche eine E-Mail-Adresse angeben. An dieser E-Mail-Adresse wird eine E-Mail verschickt, welche zur zusätzlichen Verifikation dient.

## 2.2.4 Nicht-Ziele

* Grundkosten

Das Spiel wird keine Grundkosten besitzen die man bezahlen muss um das Spiel spielen zu dürfen. Es ist für jeden frei erreichbar und spielbar.

* Keine Ablenkung vom Hauptkonzept

Es wird keine kleineren minispiele im Projekt geben die von dem Grundkonzept ablenken könnten.

# 2.3 Festlegen der Hauptdaten

* ***/LD10/Benutzerdaten:***

Zu jedem Benutzer müssen Login-Daten und Spielergebnisse gespeichert werden. Dazu wird eine Datenbank benutzt.

# 2.4 Festlegen der Hauptleistungen

* ***/LL10/Serverleistung:***

Der Server muss mehrere hundert Personen gleichzeitig verkraften können. Die Mindestanzahl die verkraftbar sein muss ist 100.

* ***/LL20/Plattformunabhängigkeit:***

Das Spiel muss von allen gängigen Betriebssystemen spielbar sein.

# 2.5 Festlegen der wichtigsten Qualitätsmerkmale

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Produktqualität** | **Sehr Gut** | **Gut** | **Normal** | **Irrelevant** |
| Funktionalität | X |  |  |  |
| Zuverlässigkeit |  |  | x |  |
| Benutzbarkeit |  |  | x |  |
| Effizienz |  |  | x |  |
| Änderbarkeit |  |  |  | x |
| Übertragbarkeit | X |  |  |  |

* **Funktionalität**

Die Applikation soll eine sehr gute Funktionalität besitzen, damit es den Benutzer anspricht, sodass der Benutzer das Spiel gerne spielt.

* **Benutzbarkeit**

Die Benutzbarkeit ist neben der Funktionalität das wichtigste bei einer Applikation da, ein kompliziertes und schwer zu spielendes Spiel schnell den Benutzer die Lust zum spielen raubt.

3. Durchführbarkeitsuntersuchung

# 3.1 Prüfen alternativer Lösungsvorschläge

## 3.1.1 Programmiersprachen/Skriptsprachen

**JAVA**

Java ist eine objektorientierte Programmiersprache.

Als IDE verwenden wir Eclipse.

Folgende Tools werden benötigt:

* Entwicklungsumgebung JDK (Java Development Kit)
* Laufzeitumgebung JRE (Java Runtime Environment)

JDK wird für das Coden von Prgrammen verwendet. JRE wird für das Ausführen von Programmen verwendet.

**Eigenschaften von Java:**

* objektorientiert
* dynamisch
* übertragbar
* robust und sicher
* einfach

**Robustheit:**

Viele der Designentscheidungen bei der Definition von Java reduzieren die Wahrscheinlichkeit ungewollter Systemfehler; zu nennen sind die starke Typisierung, Garbage Collection, Ausnahmebehandlung sowie Verzicht auf Zeigerarithmetik.

**Einfachheit:**

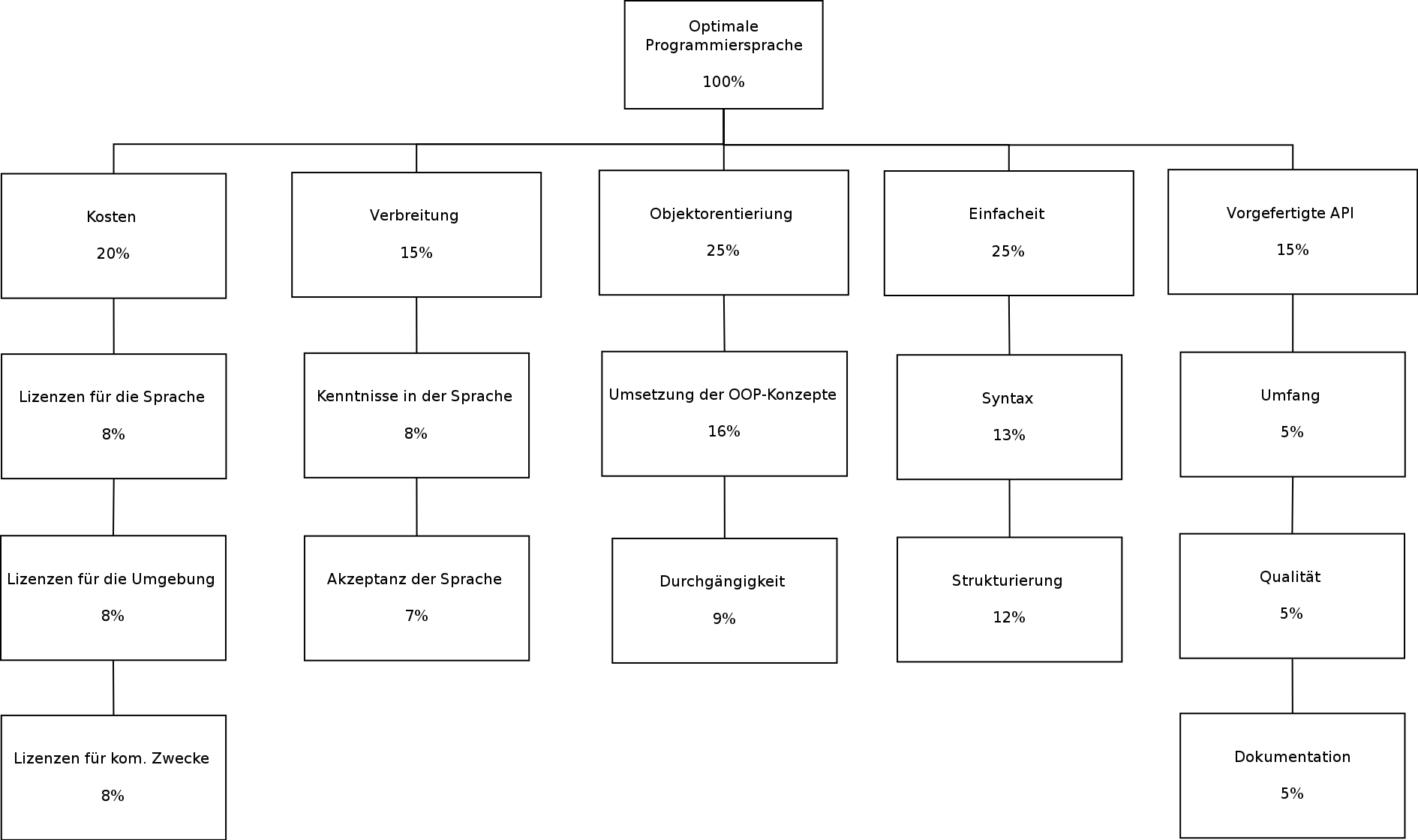
Java ist im Vergleich zu anderen objektorientierten Programmiersprachen wie C++ oder C# einfach, da es einen reduzierten Sprachumfang besitzt und beispielsweise Operatorüberladung und Mehrfachvererbung nicht unterstützt.

**Sicherheit:**

Dafür stehen Konzepte wie der Class-Loader, der die sichere Zuführung von Klasseninformationen zur Java Virtual Machine steuert, und Security-Manager, die sicherstellen, dass nur Zugriff auf Programmobjekte erlaubt wird, für die entsprechende Rechte vorhanden sind.

***Nutzwertanalyse***

Zielhierarchie:

****

Erläuterung der Kriterien:

**• Kosten**

- Lizenzen für die Sprache:

Muss für den Einsatz der Sprache eine Gebühr bezahlt werden?

- Lizenzen für die Entwicklungsumgebung:

Muss für den Erwerb bzw. Einsatz einer Entwicklungsumgebung für die jeweilige Sprache eine Gebühr bezahlt werden?

- Lizenzen für kommerzielle Zwecke:

Muss für den kommerziellen Einsatz eines in einer bestimmte Programmiersprache entwickelten Softwareprodukts eine Gebühr bezahlt werden?

**• Verbreitung**

- Kenntnisse in der Sprache

Beschreibt inwiefern unsere Entwickler bereits mit der jeweiligen Programmiersprache vertraut sind.

- Akzeptanz der Sprache

Beschreibt inwiefern unsere Entwickler die jeweilige Sprache gutheißen.

**• Objektorientierung**

- Umsetzung der OOP-Konzepte

Wurden alle Konzepte der OOP (Abstraktion, Datenkapselung, Vererbung, Polymorphie, etc.)

vollständig umgesetzt?

- Durchgängigkeit

Ist es möglich bzw. wie weit ist es möglich vom Konzept der OOP abzuweichen?

**• Einfachheit**

- Syntax

Ist die Syntax einfach, intuitiv und logisch aufgebaut?

- Strukturierung

Bietet die Programmiersprache die Möglichkeit einer einfachen und übersichtlichen Strukturierung?

**• Vorgefertigte API**

- Umfang

Wie viele vorgefertigte Klassen bzw. Programmteile stehen zur Verfügung?

- Qualität

Ist die API von einer für uns brauchbaren Qualität (d.h. fehlerfrei, effizient, etc.)?

- Dokumentation

Sind alle Inhalte der API vollständig dokumentiert?

🡪

**Fazit**

Wie aus der Nutzwerkanalyse hervorgeht, schneidet Variante 3 (HTML/PHP/Java-Script) am besten ab. Die ausschlaggebenden Gründe dafür sind vor allem die umfangreichen API’s der Programmier- bzw. Skriptsprachen und  
Das Ergebnis der Nutzwerkanalyse deckt sich zusätzlich weitgehend mit den Wunsch-Sprachen unser Entwickler, da diese bereits langjährige Erfahrungen mit diesen sammeln konnten und somit umfangreiche Kenntnisse der API’s mitbringen.

### 3.1.2 Datenbank Management System (DBMS)

**Variante 1: MySQL**

MySQL ist ein Open Source DBMS (Datenbank Management System), welches sich vor allem in Kombination mit dem Apache Webserver und der Skriptsprache PHP wachsender Beliebtheit erfreut. MySQL bietet alle Funktionalitäten, die für die Realisierung der Web-Applikation benötigt werden.

MySQL steht jedoch ausschließlich für nicht kommerzielle Zwecke kostenlos zur Verfügung.

**Variante 2: Oracle Database**

Oracle Database ist ein kostenpflichtiges DBMS des Herstellers Oracle und wird vor allem in großen Firmen eingesetzt. Oracle Database bietet weit aus mehr Funktionen als jedes andere DBMS und ist dementsprechend kompliziert zu bedienen.

Der Oracle Database Server in der aktuellen Version unterstützt 8 Millionen Terabyte (8 Exabyte) in einer Datenbank und weist damit eine Speicherkapazität auf, die derzeit (2013) kaum von einzelnen Organisationen ausgeschöpft werden kann.

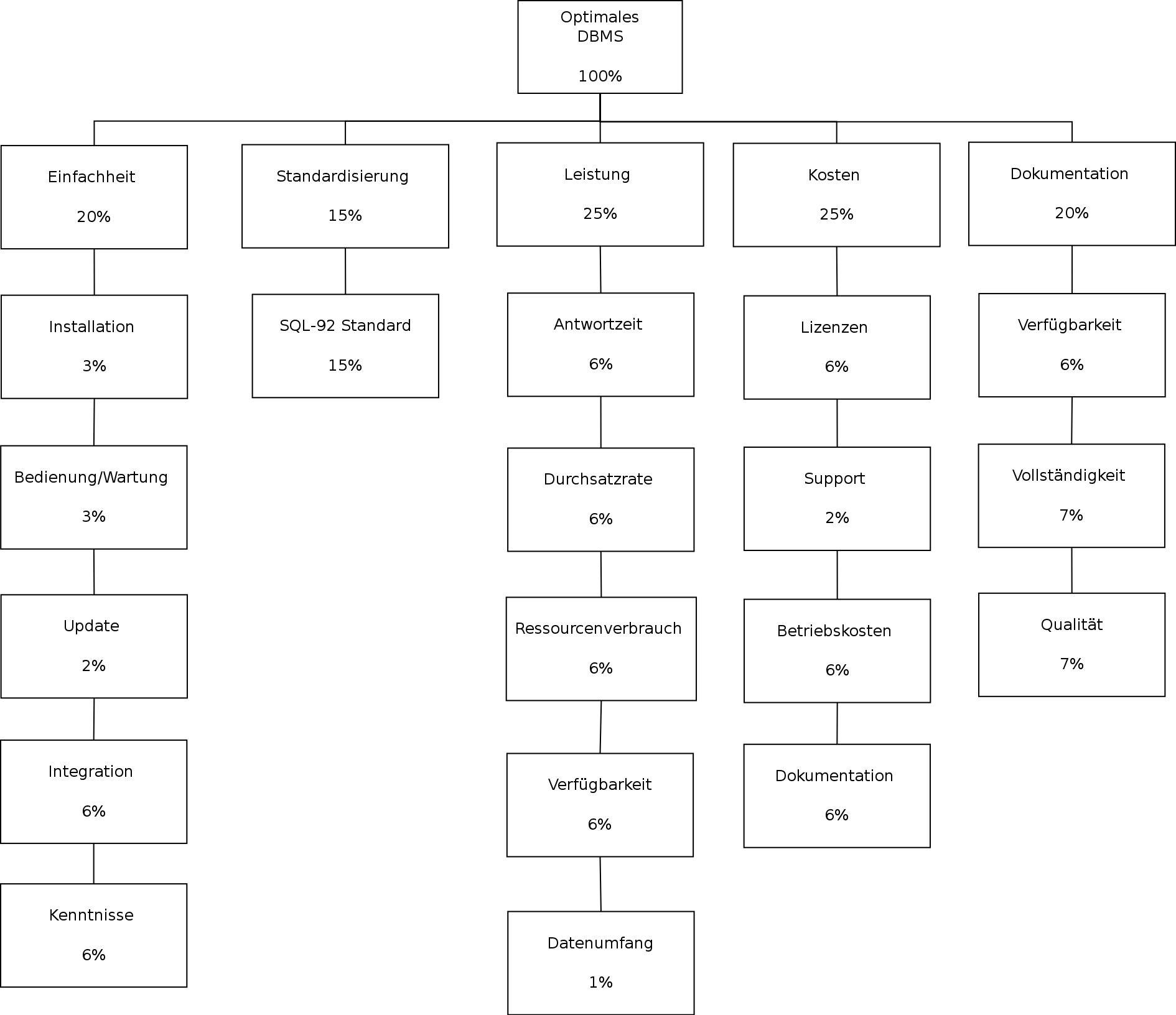
**Variante 3: PostgreSQL**

PostgreSQL ist eines der ältesten und am weitesten fortgeschrittenen ORDBMS (objektrelationalen Datenbankverwaltungssysteme) im Bereich der freien Software.

PostgreSQL ist weitgehend konform mit dem SQL92/SQL99/SQL2003-Standard, d.h. alle in dem Standard geforderten Funktionen stehen zur Verfügung und verhalten sich so wie vom Standard gefordert. Dies ist bei manchen kommerziellen sowie nichtkommerziellen SQL-Datenbanken bisweilen nicht gegeben.

PostgreSQL ist im Gegensatz zu MySQL auch im kommerziellen Einsatz frei zu verwenden.

***Nutzwertanalyse***

Zielhierarchie:

Erläuterung der Kriterien:

**• Einfachheit**

- Installation

- Bedienung/Wartung

- Update

- Integration

- Kenntnisse

**• Standardisierung**

- SQL-92 Standard

**• Leistung**

- Antwortzeit  
 Wie lange benötigt das DBMS um auf eine Abfrage zu reagieren?

- Durchsatzrate

- Ressourcenverbrauch

- Verfügbarkeit

- Datenumfang  
 Welche maximale Datenmenge kann das DBMS verwalten?

**• Kosten**

- Lizenzen  
 Muss für den Einsatz des DBMS eine Gebühr bezahlt werden?

- Support  
 Wie viel kostet Support für das jeweilige DBMS?

- Betriebskosten  
 Wie viel kostet der Betrieb des DBMS?

- Dokumentation  
 Muss für die Dokumentation des DBMS eine Gebühr bezahlt werden?

**• Dokumentation**

- Verfügbarkeit  
 Gibt es ausreichend verfügbare Dokumentationen zu dem jeweiligen  
 DBMS?

- Vollständigkeit  
 Sind verfügbare Dokumentationen vollständig?

- Qualität  
 Sind die verfügbaren Dokumentationen von einer für und  
 ausreichenden Qualität?

**Fazit**

Die für die Auswahl des Datenbank Management Systems durchgeführte Nutzwertanalyse spricht für den Einsatz von MySQL. Gründe hierfür sind einerseits die Einfachheit, andererseits auch die Häufigkeit am Markt.

Die eingeschränkte Funktionalität im Vergleich zu Oracle Database kann dadurch akzeptiert werden, da MySQL alle von uns benötigten Funktionen abdeckt.

## 3.1.3 Gesamtfazit

Nach insgesamt zwei Nutzwertanalysen steht folgendes Ergebnis fest:

1. Das Browsergame wird in HTML, Java/JSP und PHP entwickelt.

2. Die Software greift auf eine MySQL Datenbank zu.

# 3.2 Prüfen der technischen Durchführbarkeit

## 3.2.1 Technologien / Softwaretechnische Durchführbarkeit

Das Browsergame wird in HTML, PHP und Java Script entwickelt. Der Grund hierfür ist, dass diese Variante

* Einfach
* objektorientiert
* Verteilt
* robust
* sicher
* Plattform unabhängig
* portabel
* leistungsfähig
* und dynamisch ist.

Die Plattform Unabhängigkeit bietet den Vorteil, dass das Browsergame zu verschieden Betriebssystemen kompatibel ist. Das heißt, dass die Software unter anderem auf Windows, Linux und OS Systemen lauffähig sein wird.

Die objektorientierte Programmierung erlaubt es Änderungen und Weiterentwicklungen der Software in naher Zukunft effizient und fehlerfrei abzuwickeln.

PHP ist sicher. Anwendungen, die auf dieser Programmiersprache basieren, zählen zu den sichersten auf der ganzen Welt.

Die Entscheidung zu für die Programmiersprachen/Skriptsprachen werden besonders dadurch gestützt, dass bereits seit über einem Ausbildungsjahr angewendet werden.

Insgesamt kann gesagt werden, dass die Entwickler von PHP, HTML als auch Java Script eine hervorragende Basis für die Entwicklung der Software bereitstellen.

HTML,PHP und Java Script sind somit die richtige Wahl für dieses Projekt.

## 3.2.2 Technologien / Hardwaretechnische Durchführbarkeit

Da dieses Projekt keine speziellen hardwaretechnischen Ansprüche stellt, kann die hardwaretechnische Durchführbarkeit entfallen und muss nicht geprüft werden.

## 3.2.3 Verfügbarkeit von Entwicklungs- und Zielmaschinen

Das Produkt wird auf handelsüblichen Systemen mit dem Betriebssystem Microsoft Windows 7 entwickelt. Eines der verwendeten   
Entwicklungsmaschinen läuft auf dem Apple Betriebssystem Mac OS 10.7.

Die Software-Umgebung setzt sich hauptsächlich aus einem Apache Server mit PHP Interpreter und einer Anbindung zu einer MySQL-Datenbank zusammen. Adobe Dreamweaver, Adobe Photoshop, Adobe Illustrator Maxon Cinema 4D zählen zu der Entwicklungsumgebung, welche wiederum auch zu Software Umgebung zählt.

Da es sich bei diesem Software-Entwicklungsprojekt um eine Client- /Server-Applikation handelt, muss bei den Zielmaschinen in zwei Kategorien differenziert werden. Einerseits muss ein zentraler Webserver zur Verfügung gestellt werden und auf der anderen Seite befinden sich die zahlreichen Client-PCs.

Die Systemleistung des Servers muss den Mindestvoraussetzungen von WAMP,XAMP oder auch MAMP entsprechen.

Bei den Zielmaschinen genügen handelsübliche Computer, auf denen der Internet Explorer in der aktuellsten Version und oder der Mozilla Firefox Browser ebenso in der aktuellsten Version installiert ist.

# 3.3 Prüfen der personellen Durchführbarkeit

## 3.3.1 Qualifikation der Fachkräfte

Da die Projektmitglieder über das nötige technische Know-How verfügen und sehr professionell und motiviert an die Sache herangehen, stellt es kein Problem dar das Produkt zu entwickeln. Dieses Know-How wurde in der bisherigen Ausbildung sowohl theoretisch als auch praktisch oder durch freiwillige Wissensbegierde erworben.

Allerdings bestehen auch einige Risiken wie das Auftreten von unerwarteten, technischen Problemen (Ausfall von Entwicklungsmaschinen), welche die personelle Durchführbarkeit beeinflussen kann.

## 3.3.2 Zusammenarbeit der Teammitglieder

Die teaminterne Arbeit wird in diesem Projekt eine größere Rolle spielen. Da speziell in diesem Projekt zwei verschiedene Interessengruppen mit verschiedenen Kenntnissen aufeinander treffen, ist die Gefahr von Problemen relativ hoch. Gerade diese Tatsache könnte teamintern einerseits Verständigungsprobleme hervorrufen und andererseits zu einer nicht gerechten und gezielten Verteilung der Arbeitspakete führen und in Folge dessen einen Verzug innerhalb des Projektes verursachen.  
Da jedoch beide Gruppen geteilte und voneinander nicht zwingend abhängige Arbeitsbereiche besitzen, können somit diese resultierenden Risiken weitgehend vermieden werden.

Gruppenintern wurden erst wenige Erfahrungen bezüglich der Zusammenarbeit an einem gemeinsamen Projekt gesammelt. Jedoch wurden technische Probleme größtenteils durch die gute Kommunikation bereits am Ansatz behoben. Und falls es doch einmal zu einem gröberen Problem gekommen ist, half das spezielle Expertenwissen jedes einzelnen Mitarbeiters zu einer schnellen, effizienten und qualitativ hochwertigen Lösung. Alle Projektmitarbeiter waren sich von Anfang an streng bewusst, dass nur durch eine starke Zusammenarbeit große Ziele erreicht werden können.  
  
Diese bisherig gut funktionierende Zusammenarbeit wird sich zumindest gruppenintern zwischen den einzelnen Teammitgliedern in dem neuen Projekt als sehr vorteilhaft und fördernd erweisen.

# 3.4 Prüfen der Risiken

## 3.4.1 Personelle Risiken

**Ausfall eines Teammitglieds**Der Ausfall eines Teammitglieds, sei es durch Krankheiten oder andere soziale Faktoren, kann das Gelingen des gesamten Projekts gefährden. Die Risiken liegen hierbei vor allem bei der zeitgerechten Abwicklung des Projekts und auch beim möglichen Verlust von fachspezifischen Wissensquellen.

**Streit im Team**

Es gibt eine Menge Faktoren, die einen Streit innerhalb eines Teams auslösen können wie etwa Meinungsverschiedenheiten, soziale Spannungen, Unzufriedenheiten, etc., die im schlimmsten Fall auch zur Auflösung des Teams führen können. Wichtig hierbei sind die Fähigkeiten des Projektleiters, der derartige Probleme frühzeitig erkennen und unter Einsatz seiner Soft-Skills lösen muss.

## 3.4.2 Technische Risiken

**Zu hoch gesteckte Ziele/ Zu geringes technisches Know-How**Aufgrund der hohen technischen Komplexität besteht die Gefahr, dass technische Probleme auftreten, die über das technische Know-How des Projektteams hinausgehen.

Die Risiken werden hierbei durch fundierte Grundkenntnisse in den einzelnen Teilbereichen sowie bereits gesammelte Erfahrungen im Umgang mit den gewählten Technologien minimiert.

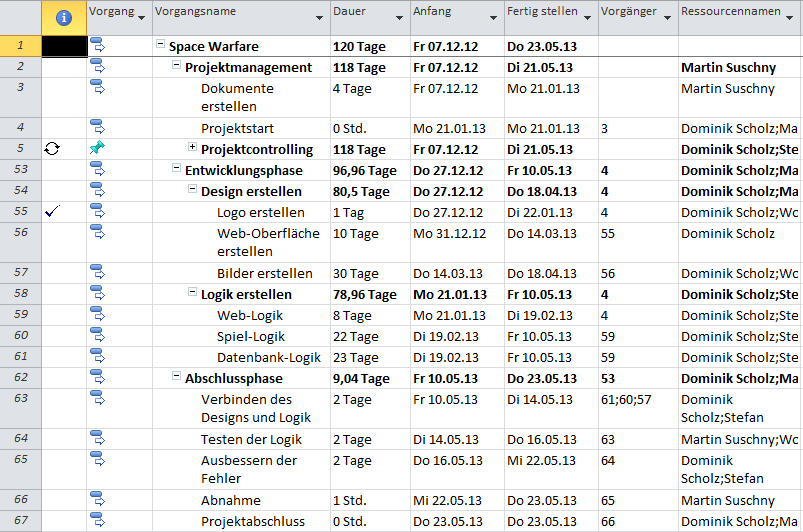
**Cross-Browser Anwendungen**  
Gerade bei Technologien wie AJAX (JavaScript, CSS, HTML) in Verbindung mit einer Webprogrammiersprache könnte es zu Browserinkompatibilitäten kommen. Hierbei müsste man jede Anwendung Cross-Browser fähig machen, welches bei größeren Tools zu sehr viel Aufwand führen würde und somit ein direktes Risiko für die Termintreue darstellt.

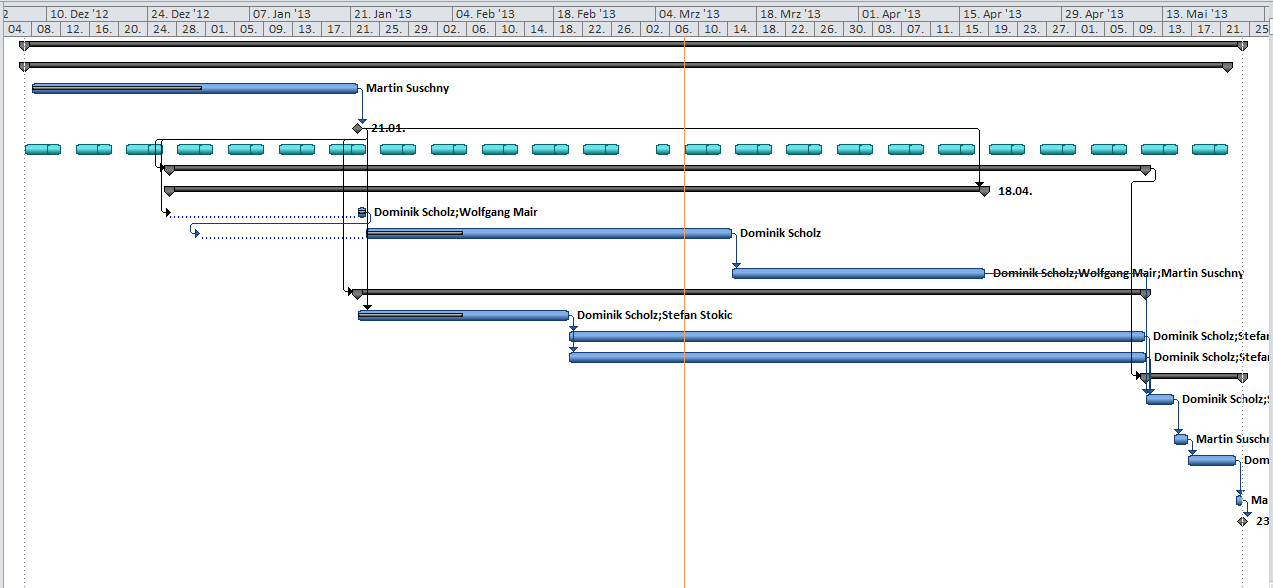
Dieses Risiko kann durch eine klare Abgrenzung von Browsern, unter welchen die Web-Applikation laufen wird, behoben werden.

Um die Umsetzung von Cross-Browser-fähigen Applikationen in einem realistischen Rahmen zu halten, muss die Unterstützung gewisser Browser eingeschränkt werden, unter der Bedingung der Abdeckung eines prozentuell überwiegenden Spektrums an Benutzern am Markt. Die Vernachlässigung kaum populärer Browser muss hierbei in Kauf genommen werden um den Aufwand innerhalb des Projektes in Grenzen zu halten.  
  
**Ansprüche und Standards im Web**Da Heutzutage die Standards und Anforderungen einer Webapplikation im kommerziellen Bereich sehr hoch angesetzt sind, wird es sicherlich nicht allzu einfach sein den Ansprüchen der heutigen Zeit gerecht zu werden, vor allem wenn es um zukunftsweisende Technologien handelt, da solche immer mit Vorsicht zu genießen sind.  
  
Durch mangelnde Definitionen, Richtlinien und Abgrenzungen fehlt meist die nötige Orientierung im Umgang mit neuen Technologien. Die Berücksichtigung und Einhaltung von Standards stellt zeitlich gesehen ein Risiko für die Termintreue und somit für den Projekterfolg dar.  
  
Weiters setzen sich heutzutage definierte Standards und Kreativität gegenseitig Grenzen. Genau hier muss darauf geachtet werden, dass bei der Entwicklung einer Software keine schwerwiegenden Probleme entstehen.  
  
Um nun eine gewisse Qualität und Standards zu gewährleisten, müssen diverse Funktionalitäten in ihrer Komplexität der Umsetzung eingeschränkt werden.

# 3.5 Prüfen der ökonomischen Durchführbarkeit

## 3.5.1 Terminschätzung

**Allgemein**Die regelmäßige und rechtzeitige Rückmeldung der Termine an unseren Projektleiter ist die Voraussetzung für eine wirkungsvolle Terminkontrolle.



**Meilensteine**

|  |  |
| --- | --- |
| **Meilenstein** | **Datum** |
| **Projektstart** | 7.12. 2013 |
| **Abschluss des Vorprojekts** | 21.01.2013 |
| **Abschluss der Entwicklungsphase** | 10.05.2013 |
| **Abschluss der Bugfixes** | 22.05.2013 |
| **Projektabnahme** | 22.05.2013 |

**Terminverschiebungen**

Die Gründe für eine mögliche Terminverschiebung in unserem Projekt

werden durch folgende Faktoren bestimmt:

• Technische Probleme

• Personalengpässe

I. Krankheit von Mitgliedern

II. Ausfall von Mitgliedern

• Geringe Produktivität des Projektteams

I. Schlechte Koordination

II. Mangelhafte Sachkenntnisse

Sollte es zu einer Terminverschiebung kommen, erfolgt eine sofortige

Rückmeldung an unseren Projektleiter, um dementsprechende Maßnahmen zu ergreifen.

**Maßnahmen bei Terminproblemen**

Die Maßnahmen bei einem möglichen Terminproblem werden nur nach Berücksichtigung aller Rückmeldungen unserer Teammitglieder erfolgen. Mögliche Maßnahmen für den Fall, dass ein Termin nicht eingehalten werden kann, sind:

* Erhöhung der Arbeitszeit   
  I. Überstunden
* Erhöhung der Produktivität   
  I. Optimierung der Arbeitsabläufe

Zusätzliche Personalressourcen

4.Nutzenanalyse

# 4.1 Nutzen für den Kunden

Dem Kunden wird eine schnelle und einfache Unterhaltung für den Alltag durch den Browser ermöglicht.   
  
4.2 Nutzen für das Projektteam

Das Projekt wird auf eindrucksvoller Weise auf die Fähigkeiten des Teams aufmerksam machen. Außerdem wird es dem Projektteam den späteren Einstieg in andere Firmen erleichtern, da das Projektteam bei der Bewerbung das Projekt präsentieren kann.