

Fachinformatiker für Anwendungsentwicklung

Dokumentation zur schulischen Projektarbeit

**Verwaltungssoftware**

für das Fitnessstudio FIT

Abgabetermin: 11.07.2018

**Projektgruppe:**

Deniel Balic, Groz-Beckert KG

Jens Dreher, Gühring KG

Laura Knupfer, Groz-Beckert KG

Jonas Reinecke, yourIT GmbH

**Inhaltsverzeichnis**

1. **Einleitung Seite | 3**
   1. Projektbeschreibung Seite | 3
   2. Projektziel Seite | 3
   3. Projektumfeld Seite | 3
   4. Projektbegründung Seite | 3
   5. Projektschnittstellen Seite | 4
   6. Projektabgrenzung Seite | 4
2. **Projektplanung Seite | 4**
   1. Projektphasen Seite | 4
      1. Projektstrukturplan Seite | 4
   2. Ressourcenplanung Seite | 5
   3. Programmiersprache Seite | 5
3. **Analysephase Seite | 5**
   1. Ist-Analyse Seite | 5
   2. Wirtschaftlichkeitsanalyse Seite | 6
      1. „Make or Buy“ – Entscheidung Seite | 6
      2. Projektkosten Seite | 6
      3. Kosten-Nutzen-Analyse Seite | 6
   3. Lastenheft Seite | 7
4. **Projektdurchführung Seite | 7**
   1. Angebot und Projektauftrag Seite | 7
   2. Pflichtenheft Seite | 7
   3. Benutzeroberfläche Seite | 8
   4. Programmierung Seite | 8
      1. Mitgliederverwaltung Seite | 8
      2. Mitarbeiterverwaltung Seite | 9
      3. Terminal Seite | 9
      4. Kursplan Seite | 9
      5. Registrierung und Login Seite | 10
   5. Anpassungen Seite | 10
5. **Arbeitsprotokolle Seite | 10**
6. **Projektergebnisse Seite | 11**
   1. Soll-/Ist-Vergleich Seite | 11
   2. Abnahme durch den Auftraggeber Seite | 11
   3. Lessons Learned Seite | 11
   4. Ausblick Seite | 11

**Anhänge Seite | 12**

* 1. Gantt-Diagramm Seite | 12
  2. Projektstrukturplan Seite | 13
  3. Verwendete Ressourcen Seite | 13
  4. Lastenheft Seite | 14
  5. Angebot und Projektauftrag Seite | 17
  6. Pflichtenheft Seite | 19
  7. Aufgabenverteilung Seite | 21
  8. Arbeitsprotokolle Seite | 22

1. **Einleitung**

Die folgende Projektdokumentation beschreibt den Ablauf des Schulprojektes, das während des zweiten Lehrjahres der Ausbildung zum Fachinformatiker mit Fachrichtung Anwendungsentwicklung umgesetzt wurde. Die Umsetzung erfolgte durch eine Gruppe von vier Schülern, die jeweils in einem anderen Ausbildungsbetrieb beschäftigt sind.

* 1. **Projektbeschreibung**

Die Projektidee entstand aufgrund der aktuellen Mitgliederverwaltung. Bisher werden alle mitgliederbezogenen Daten noch in Papierform verwaltet, was sehr viel unnötige Zeit für die Trainer des Fitnessstudios FIT in Anspruch nimmt. Ihre Hauptaufgabe sollte eigentlich in der Mitgliederbetreuung und Kursgebung liegen. Außerdem fehlt die Möglichkeit zu sehen, welche Mitglieder aktuell im Studio trainieren.

Da die Digitalisierung auch die Branche Fitness schon lange erreicht hat, besteht hier unbedingt Handlungsbedarf, um den Schritt der Digitalisierung nicht zu verpassen.

* 1. **Projektziel**

Das Ziel des Projektes ist das Erschaffen einer übersichtlichen Verwaltungssoftware für das Fitnessstudio FIT, die eine leichte Bedienung für die Mitarbeiter ermöglicht.

Zu der Software gehört eine Mitgliederdatenverwaltung, eine Mitarbeiterdatenverwaltung, die Kurspläne, das Terminal, sowie die Registrierung und das Login. Diese Funktionen werden im Lastenheft genauer beschrieben.

* 1. **Projektumfeld**

Auftraggeber des Projektes ist das Fitnessstudio FIT in Musterhausen, das 20 Mitarbeiter in Teilzeit beschäftigt und derzeit rund 500 Mitglieder betreut.

Zwei der zwanzig Mitarbeiter kümmern sich um die Buchhaltung und die Verwaltung der Mitarbeiterdaten.

Die anderen Mitarbeiter, das heißt die Fitnesstrainer, kümmern sich hingegen um das Einführen neuer Trainingspläne, die Annahme neuer Mitglieder, um Beratungsgespräche, sowie um die Kursgebung und das Verwalten der Mitgliederdaten. Zusätzlich schreiben sich die Trainer händisch auf, wenn ein Mitglied zum Trainieren kommt, beziehungsweise wieder geht. Diese Aufschriebe kommen oftmals abhanden, was dazu führt, dass man nie ganz sicher weiß, welches Mitglied gerade trainiert. Dies kann im Falle eines Brandes zu einer großen Gefahr werden.

Um den Anforderungen des Fitnessstudios gerecht zu werden, ist eine intensive und regelmäßige Kommunikation und Rücksprache mit dem Auftraggeber erforderlich.

* 1. **Projektbegründung**

Die Hauptschwachstelle der Arbeit des Fitnessstudios ist das hohe Maß an manueller Arbeit für die Nebenaufgaben. Dazu gehören die Mitgliederverwaltung und das schriftliche Erfassen der aktuell trainierenden Mitglieder. Diese beanspruchen sehr viel Zeit und lenken die Mitarbeiter im Außenbereich von ihren Hauptaufgaben ab.

Ein weiteres Problem ist, dass für 500 Mitglieder alle Unterlagen in Papierform aufbewahrt werden, was viel physischen Raum in Anspruch nimmt und somit Kosten verursacht. Dies kann durch eine digitale Datenverwaltung vollständig vermieden werden.

* 1. **Projektschnittstellen**

Damit die Mitgliederdaten und Mitarbeiterdaten von der Software verwaltet werden können, wird eine Datenbank benötigt. Diese beinhaltet die persönlichen Daten, sowie verschiedene Dokumente, wie zum Beispiel die Verträge, Scans oder ähnliches. Hierfür wird eine MySQL Datenbank verwendet, die an die WPF-Anwendung, mit der die Software geschrieben wird, angebunden ist.

Diese MySQL-Datenbank wird mithilfe von XAMPP verwaltet. XAMPP ist eine webbasierte Testserverumgebung, die alle wichtigen Komponenten zum Betreiben einer MySQL-Datenbank enthält. Das heißt, die Funktion der Datenbank kann mit dieser Anwendung getestet und konfiguriert werden.

Benutzt wird die Anwendung ausschließlich von den Mitarbeitern des Fitnessstudios FIT.

* 1. **Projektabgrenzung**

Da der Projektumfang beschränkt ist, soll das Bereitstellen eines integrierten Terminkalenders, sowie das automatische Erstellen von Statistiken nicht Bestandteil des Schulprojektes sein.

1. **Projektplanung** 
   1. **Projektphasen**

Für die Umsetzung des Projektes standen den Schülern 70 Stunden zur Verfügung. Diese wurden auf verschiedene Phasen aufgeteilt, die während der Entwicklung durchlaufen werden. Eine grobe Zeitplanung, sowie die Hauptprojektphasen können der Tabelle 1: Grobe Zeitplanung entnommen werden. Die Hauptphasen wurden wiederum in kleinere Unterpunkte beziehungsweise Arbeitspakete zerlegt. Eine detaillierte Übersicht dieser Phasen befindet sich im Anhang A.1: Gantt-Diagramm auf S. 12.

|  |  |
| --- | --- |
| **Projektphase** | **Geplante Zeit** |
| Analysephase | 5 h |
| Durchführung | 45 h |
| Abnahme und Einführung | 4 h |
| Erstellung Dokumentation | 10 h |
| **Gesamt** | **70 h** |

Tabelle 1: Grobe Zeitplanung

* + 1. **Projektstrukturplan**

Im Projektstrukturplan wurden alle zu erledigenden Aufgaben gegliedert und in einzelne Arbeitspakete zerlegt, um einen ersten Überblick über das gesamte Projekt zu erhalten. Dieser Projektstrukturplan findet sich um Anhang A.2: Projektstrukturplan auf S. 13.

* 1. **Ressourcenplanung**

Die Übersicht über die verwendeten Ressourcen, welche sich im Anhang A.3: Verwendete Ressourcen auf S. 13 befindet, zeigt alle Ressourcen, die für das Projekt eingesetzt wurden. Darunter fallen sowohl Hard- und Softwareressourcen, als auch das Personal. Um die Projektkosten möglichst gering zu halten, wurde bei der Auswahl der verwendeten Software darauf geachtet, dass diese kostenfrei (z. B. als Open Source) zur Verfügung steht.

* 1. **Programmiersprache**

Die Programmiersprache wurde anhand der Programmierkenntnisse der Entwickler der Projektgruppe festgelegt. Da alle Mitglieder in der Berufsschule die Sprachen C und C++ bereits behandelt haben, entschied sich die Gruppe für die Sprache C-Sharp (C#), da diese sehr ähnlich zu den bereits gelernten Sprachen ist. Die Gruppe entschied sich auch aus dem Grund dafür, da die Sprache zwar den bereits gelernten Sprachen ähnlich, aber nicht genau gleich ist und somit bei der Programmierung der Anwendung neue Programmiererfahrungen gesammelt werden konnten.

Letztendlich entschied man sich dann für eine Windows Presentation Foundation (WPF)-Anwendung in der Sprache C#. WPF ist eine Klassenbibliothek zur Entwicklung von graphischen Benutzeroberflächen und ist Bestandteil des .NET-Frameworks von Microsoft. Da der Kunde eine Desktop-Anwendung mit ansprechendem Design angefordert hat, erwies sich die WPF-Anwendung als beste Umsetzungsmöglichkeit für das Projekt.

Durch WPF werden Entwicklungs- und Wartungskosten reduziert, da das darstellungsspezifische Markup und der verhaltensspezifische Code nicht eng aneinandergekoppelt sind. Die Entwicklung ist effizienter, da Designer und Entwickler parallel die Darstellung der Anwendung und das Verhalten der Anwendung implementieren können.

1. **Analysephase**
   1. **Ist-Analyse**

Wie bereits im Abschnitt 1.1 (Projektbeschreibung) erwähnt wurde, werden die Trainer des Fitnessstudios durch viele Nebenaufgaben zu sehr von ihrer eigentlichen Arbeit abgelenkt.

Die erste Nebenaufgabe ist die Mitgliederverwaltung. Wenn eine Person Mitglied des Fitnessstudios werden möchte, muss sie dies an der Rezeption des Studios einem Trainer mitteilen. Anschließend gibt es ein Gespräch, bei dem geklärt wird, was für einen Vertrag die Person haben möchte und für wie lange. Anhand dieser Informationen druckt der Trainer den entsprechenden Vertrag, sowie das Formular zum Mitgliedsbeitritt aus. Unter Aufsicht des Trainers füllt das neue Mitglied das Formular aus und unterschreibt den Vertrag für das Studio und die Kopie für sich selbst. Das neue Mitglied erhält eine fortlaufende Nummer als Mitgliedsnummer und die beiden Formulare werden vom Trainer im Mitgliederordner abgeheftet. Anschließend wird alles manuell in eine Excel-Liste eingepflegt, die die persönlichen Daten der Mitglieder enthält.

Die Mitarbeiterverwaltung ist der Mitgliederverwaltung sehr ähnlich. Die Büromitarbeiter müssen alle auszufüllenden Formulare und Verträge ausdrucken und heften diese anschließend in einen Ordner ein. Zudem wird der neue Mitarbeiter ebenfalls manuell in eine Excel-Liste eingetragen.

Eine weitere Aufgabe ist der Kursplan. Dieser ist momentan seit einem Jahr unverändert und wird in Papierform ausgehängt. Kurse finden an jedem Wochentag, außer am Sonntag, statt.

Zuletzt gibt es noch das Terminal, dessen Ablauf im Abschnitt 1.3 (Projektumfeld) genau beschrieben ist.

* 1. **Wirtschaftlichkeitsanalyse**

Aufgrund der zusätzlich anfallenden Nebentätigkeiten im Fitnessstudio, die in den Abschnitten 1.4 (Projektbegründung) und 3.1 (Ist-Analyse) erläutert wurden, ist die Umsetzung des Projektes für das Fitnessstudio erforderlich. Ob die Realisierung auch aus wirtschaftlichen Gesichtspunkten gerechtfertigt ist, soll in den folgenden Abschnitten geklärt werden.

* + 1. **„Make or Buy“-Entscheidung**

Da es sich bei dem Projekt um eine sehr spezifische Anforderung des Fitnessstudios FIT handelt, die leicht zu bedienen und nicht sehr umfangreich sein soll, lässt sich auf dem Markt keine Lösung finden. Von den Kosten ist es billiger eine Anwendung entwickeln zu lassen, anstatt auf dem Markt eine Anwendung zu kaufen, die viel mehr Funktionen bietet als eigentlich gebraucht werden.

* + 1. **Projektkosten**

Die während der Entwicklung anfallenden Projektkosten werden im Folgenden kalkuliert. Neben den Personalkosten sollten hier auch die Aufwendungen für die Ressourcen (Hard- und Software) berücksichtigt werden. Der Stundensatz der Projektmitarbeiter wurde auf 15 € pro Stunde festgelegt.

Die für die einzelnen Vorgänge des Projektes anfallenden Kosten, sowie die gesamten Projektkosten können der Tabelle 2: Kostenaufstellung entnommen werden.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Dauer in h** | **Stundensatz in €** | **Kosten pro AP** |
| **Aufnahme der Kundenanforderung** | 26 | 60,00 | 1.560,00 |
| **Dienstleistung Programmierung** | 60 | 75,00 | 4.00,00 |
| **Dienstleistung Dokumentation** | 8 | 55,00 | 440,00 |
|  |  |  |  |
| **Fremdleistungen** | -- | -- | -- |
| **Materialkosten** | -- | -- | 1.500,00 |
| **Selbstkosten** |  |  | 6.380,00 |
| **Kalkulierter Gewinn** |  |  | 1.520,00 |
| **Projektgesamtkosten** |  |  | **8.000,00** |

* + 1. **Kosten-Nutzen-Analyse / Break-Even-Point**

Anhand der Kosten-Nutzen-Analyse soll ermittelt werden, ab welchem Zeitpunkt die durch die Entwicklung des Verwaltungsprogramms anfallenden Kosten durch die Einsparungen gedeckt werden. Dieser Punkt wird als „Break-Even-Point“ bezeichnet. Anhand dieser Analyse kann beurteilt werden, ob die Umsetzung des Projektes aus wirtschaftlicher Sicht sinnvoll ist und sich auf Dauer Kostenvorteile ergeben.

Im Folgenden werden zuerst die monatlichen Ausgaben des Fitnessstudios aufgeführt und anschließend die Kosten-Nutzen-Analyse bzw. der Break-Even-Point graphisch dargestellt.

|  |  |
| --- | --- |
| **Ausgaben** | **Monatliche Kosten in €** |
| Lagerkosten | 450,00 |
| Materialkosten | 25,00 |
| Verwaltungsaufwand | 250,00 |
| **Gesamt** | **725,00** |

* 1. **Lastenheft**

Das Lastenheft wurde am Ende der Analysephase zusammen mit dem Inhaber des Fitnessstudios FIT erstellt. Es beinhaltet alle Anforderungen, die an die zu entwickelnde Anwendung gestellt werden. Das Lastenheft befindet sich im Anhang A.4: Lastenheft auf S. 14.

1. **Projektdurchführung**
   1. **Angebot und Projektauftrag**

Im ersten Schritt der Projektdurchführung wurde das Angebot anhand der während der Planung erstellten Kostenkalkulation zusammengestellt. Dieses wurde anschließend dem Kunden vorgelegt und durch dessen Unterschrift angenommen. Da nun das Projekt starten konnte, wurde der Projektauftrag erstellt, der ebenfalls durch die Unterschrift des Auftraggebers, sowie des Projektleiters bestätigt wurde. Diese beiden Dokumente können im Anhang A.5: Angebot und Projektauftrag auf S. 17 begutachtet werden.

* 1. **Pflichtenheft**

Nach der Erstellung des Angebots und des Projektauftrags wurde ein Pflichtenheft erstellt. Dieses baut auf dem Lastenheft auf und beschreibt, wie und womit die Projektgruppe die Anforderungen des Fitnessstudios umsetzen möchte. Somit dient es als Leitfaden für die Realisierung des Projektes. Das Pflichtenheft befindet sich im Anhang A.6: Pflichtenheft auf S. 19.

* 1. **Benutzeroberfläche**

Bevor mit der Programmierung der Anwendung begonnen wurde, entwarf die Gruppe ein grobes Layout der Benutzeroberfläche und präsentierte dieses dem Kunden. Der Grund dafür war, dass der Auftraggeber eine leicht bedienbare und benutzerfreundliche Anwendung angefordert hatte und die Gruppe diese Anforderung auch erfüllen wollte. Gemeinsam mit den Mitarbeitern von FIT wurde dann in einem Meeting das finale Design der Anwendung festgelegt, anhand der die Entwickler nun programmieren konnten.

Dabei wurde darauf geachtet, dass nicht zu viele Funktionen zur Verfügung stehen und die Buttons für ihre Funktion sprechen. Außerdem wurde auf ein ansprechendes Design geachtet, da dies dem Auftraggeber wichtig war. Die Erstellung des Layouts und Designs nahm somit schon viel Zeit in Anspruch.

Die speziellen Design-Anforderungen der Benutzeroberfläche konnten mit der Markupsprache Extensible Application Markup Language (XAML), die Bestandteil der WPF-Anwendung ist, am besten umgesetzt werden. Mit XAML wird die Darstellung der Desktop-Anwendung implementiert und mithilfe von verwalteten Programmiersprachen (Code-Behind) die entsprechende Laufzeitlogik.

* 1. **Programmierung**

Bevor die Programmierung startete, wurde innerhalb der Gruppe festgelegt welcher Entwickler welche Funktion programmieren soll. Die Funktionen sind die Mitgliederverwaltung, die Mitarbeiterverwaltung, die Terminal, der Kursplan und zuletzt die Registrierung mit Login. Diese Verteilung kann im Anhang A.7: Aufgabenverteilung auf S. 21 eingesehen werden.

Da für das Verwaltungsprogramm eine Datenbank benötigt wurde, war es zuerst notwendig, sich über die Datenbankanbindung zu informieren. Währenddessen starteten die anderen Mitglieder mit dem groben Grundgerüst für die jeweiligen Funktionen.

Nachdem die Datenbankanbindung implementiert wurde, kümmerte sich jedes Mitglied um die Programmierung seiner Funktion. Bei Problemen halfen sich die Teammitglieder gegenseitig und trugen somit gemeinsam zum Erreichen der Fertigstellung der Anwendung bei.

Diese Arbeitsteilung war hauptsächlich durch die WPF-Anwendung möglich. Während ein Teammitglied sich um die Seitenoberfläche der jeweiligen Funktion kümmerte, konnte ein anderer Entwickler bereits mit der Code-Behind-Programmierung in C# beginnen.

* + 1. **Mitgliederverwaltung**

Mithilfe der erstellten Entwürfe bezüglich des Layouts und des Designs wurde zuerst die Benutzeroberfläche (Markup) mit XAML programmiert. Aufbauend auf die Benutzeroberfläche folgte die Entwicklung der Laufzeitlogik.

Über den Button „Alle Mitglieder anzeigen“ wurden aus der MySQL Datenbank die Daten der Mitglieder selektiert und im Grid der Mitgliederverwaltung in einer Tabelle aufbereitet.

Der Button „Mitglied suchen“ führt eine Funktion aus, dass nach Eingabe der Kundennummer der entsprechende Kunde in der rechten Tabelle angezeigt wird. Durch Auswählen des Mitglieds in der Tabelle wird ein neues Fenster geöffnet, in dem alle Informationen und Daten dargestellt werden. Wird eine fehlerhafte Kundennummer eingegeben, erscheint eine dementsprechende Fehlermeldung.

In diesem Fenster gibt es die Möglichkeit, das ausgewählte Mitglied über den Button „Mitglied löschen“ aus der Datenbank wieder zu entfernen.

Soll ein neuer Kunde hinzugefügt werden, so kann im Hauptfenster über den Button „Mitglied hinzufügen“ ebenfalls ein weiteres Fenster geöffnet werden, in das die Daten des Mitglieds eingetragen und in der Datenbank über den Button „Hinzufügen“ gespeichert werden. Hierfür müssen allerdings alle Felder ausgefüllt, sowie eine korrekte E-Mail Adresse eingegeben werden. Ist dies nicht der Fall, wird durch eine Fehlermeldung darauf hingewiesen, dass alle Felder ausgefüllt werden müssen.

* + 1. **Mitarbeiterverwaltung**

Die Mitarbeiterverwaltung wurde aufgrund der Übersichtlichkeit und der leichten Bedienbarkeit an die Verwaltung der Mitglieder angepasst.

Über den Button „Alle Mitarbeiter anzeigen“ wurden aus der MySQL Datenbank die Daten der Mitarbeiter selektiert und im Grid der Mitarbeiterverwaltung in einer Tabelle aufbereitet.

Der Button „Mitarbeiter suchen“ führt eine Funktion aus, dass nach Eingabe der Mitarbeiternummer der entsprechende Mitarbeiter in der rechten Tabelle angezeigt wird. Durch Auswählen des Mitarbeiters in der Tabelle, wird ein neues Fenster geöffnet, in dem alle Informationen und Daten dargestellt werden. Wird eine fehlerhafte Kundennummer eingegeben, erscheint eine dementsprechende Fehlermeldung.

In diesem Fenster gibt es die Möglichkeit, den ausgewählten Mitarbeiter über den Button „Mitarbeiter löschen“ aus der Datenbank wieder zu entfernen.

Soll ein neuer Mitarbeiter hinzugefügt werden, so kann im Hauptfenster über den Button „Mitarbeiter hinzufügen“ ebenfalls ein weiteres Fenster geöffnet werden, in das die Daten des Mitarbeiters eingetragen und in der Datenbank über den Button „Hinzufügen“ gespeichert werden. Hierfür müssen allerdings alle Felder ausgefüllt, sowie eine korrekte E-Mail Adresse eingegeben werden. Ist dies nicht der Fall, wird durch eine Fehlermeldung darauf hingewiesen, dass alle Felder ausgefüllt werden müssen.

* + 1. **Terminal**

Die Seite „Terminal“ dient den Mitarbeitern zur Übersicht, wie viele und welche Mitglieder aktuell im Studio trainieren.

Über den Button „CheckIn“ kann ein Mitglied eingebucht werden. Davor wird allerdings im Eingabefeld die Kundennummer eingetragen und mit der Datenbank auf Existenz gegengeprüft. Sobald der Button betätigt wird, erscheint das Mitglied in der unteren Liste, die alle anwesenden Mitglieder beinhaltet. Mit der „Buchung“ wird die aktuelle Uhrzeit gezogen und ebenfalls in der Liste aufgeführt.

Sobald das Mitglied das Studio verlässt, wird es in der Liste ausgewählt und über den Button „CheckOut“ aus der Liste wieder entfernt.

* + 1. **Kursplan**

Über den Reiter „Kursplan“ können Kurse angelegt, eingesehen und wieder gelöscht werden. Er dient zur Übersicht über alle Kurse für die Mitglieder und Mitarbeiter.

In einer Liste werden alle laufenden und bevorstehenden Kurse angezeigt. Daneben wurde ein Kalender implementiert, der eine Übersicht für den Mitarbeiter darstellt.

Wird ein neuer Kurs angeboten, so kann dieser durch den Button „Kurs hinzufügen“ neu angelegt werden. Ein neues Fenster öffnet sich, in dem der Kursname, der Tag, die Uhrzeit und die Anzahl der Teilnehmer des Kurses angegeben werden. Wird der Kurs über den entsprechenden Button hinzugefügt, wird er in die Datenbank geschrieben und ebenfalls in der Kursübersicht angezeigt.

Über den Button „Kurs absagen“ im Hauptfenster können angebotene Kurse abgesagt und wieder aus der Liste entfernt werden.

* + 1. **Registrierung und Login**

Beim Start der Anwendung erscheint ein Registrierungsfenster zur Mitarbeiterregistrierung. Hier werden neue Mitarbeiter für das Verwaltungsprogramm angelegt. Dabei muss der zukünftige Mitarbeiter den Vornamen, den Nachnamen, seine Adresse, eine gültige E-Mail Adresse, sowie ein Passwort und dessen Wiederholung eingeben. Über den Button „Registrieren“ werden die Eingaben überprüft und in die Datenbank geschrieben. Die Mitarbeiternummer wird währenddessen automatisch vergeben. Das heißt, die Mitarbeiternummern werden fortlaufend von 001 aus hochgezählt und jeweils die nächste freie Nummer vergeben.

Bereits registrierte Mitarbeiter können über den Link „Login“ direkt zum Anmeldefenster wechseln. Hier wird die E-Mail und das Passwort eingegeben und mit der Datenbank gegengeprüft. Bei richtiger Eingabe erfolgt die Anmeldung über den Button „Login“ und das eigentliche Programm wird gestartet.

Alle Mitarbeiter besitzen im Programm dieselben Administrationsrechte. Der jeweils angemeldete Mitarbeitername wird aus der Datenbank gezogen und oben links angezeigt.

Jeder User hat die Möglichkeit, über den Button „Abmelden“ oben rechts zurück zum Anmeldefenster zu navigieren oder über das Ausschaltsymbol das Programm zu beenden und automatisch abgemeldet zu werden.

* 1. **Anpassungen**

Im Folgenden werden die vorgenommenen Anpassungen während der Projektdurchführung kurz erläutert.

Anfangs war geplant, für die Datenbank die MySQL-Konsolenanwendung zu verwenden. Im Laufe der Durchführung hat sich das Team jedoch für XAMPP entschieden, da die Datenbanken dort einfacher einzusehen und zu verwalten sind.

Die anfangs geplante Zeitbuchung wurde in die Funktion „Terminal“ umbenannt. Der Grund dafür ist, dass es dem Fitnessstudio ausreicht, die aktuell trainierenden Mitglieder mit der Anfangszeit zu sehen. Somit entfällt die bisherige „Zeitbuchung“, da es nicht notwendig ist, die Trainingsdauer aufzuzeichnen, sondern nur eine Übersicht über die aktuell trainierenden Mitglieder vorhanden sein soll.

1. **Arbeitsprotokolle**

Die Arbeitsprotokolle der Mitglieder wurden täglich erstellt. Dadurch konnte der aktuelle Stand der Projektdurchführung nachverfolgt werden und gegebenenfalls festgestellt werden, für welche Aufgaben doch mehr Zeit als ursprünglich festgelegt eingeplant werden muss.

Zudem dienten diese Arbeitsprotokolle dem Auftraggeber zur Überprüfung der Einhaltung der Fertigstellung des Projektes und zum Abrufen von Zwischenständen.

Diese Arbeitsprotokolle befinden sich im Anhang A.8: Arbeitsprotokolle auf S. 22.

1. **Projektergebnisse**
   1. **Soll-/Ist-Vergleich**

Im Rückblick auf das Projekt kann festgehalten werden, dass alle im Pflichtenheft genannten Anforderungen eingehalten wurden. Zwar wurde die Funktionalität der Zeitbuchung bzw. des Terminals geändert, aber dies wirkte sich nicht negativ auf die Erfüllung der Anforderungen aus.Die eingeplante Zeit der verschiedenen Projektphasen konnte nur teilweise realisiert werden, da beim Projektbeginn der tatsächliche Aufwand schwer einzuschätzen war. Somit wurde geringfügig vom Zeitplan abgewichen, aber dennoch konnte der festgelegte Übergabezeitpunkt eingehalten werden.

* 1. **Abnahme durch den Auftraggeber**

Nachdem die Anwendung fertiggestellt wurde, konnte diese dem Kunden präsentiert werden. Das Verwaltungsprogramm für das Fitnessstudio FIT wurde dessen Besitzer und Mitarbeiter zuerst vorgestellt. Dabei wurden alle Funktionen, sowie die Kosteneinsparungen und der Projektablauf erläutert. Anschließend folgte ein Workshop für die Anwender, in dem alle Anwendungsfälle des Programms simuliert und geschult wurden. Die Mitarbeiter konnten das Programm dabei selbst testen und offene Fragen konnten beantwortet werden.

Die Verwaltungssoftware wurde vom Auftraggeber, das heißt dem Inhaber des Fitnessstudios, mit Zustimmung der Mitarbeiter abgenommen. Somit konnte der Installationsvorgang erfolgen. Dabei wurde den Mitarbeitern gezeigt, wie sich das Programm auf ihrem Computer starten lässt und was sie in Fällen eines Programmausfalls zu tun haben.

* 1. **Lessons Learned**

Während der Durchführung des Projekts konnten die Autoren neue Erkenntnisse und Erfahrungen über die Planung und Durchführung von Projekten sammeln. Dabei wurde ersichtlich, wie wichtig eine regelmäßige Kommunikation untereinander für den Beitrag zum Gruppenerfolg ist.

Außerdem wurden neue Erkenntnisse im Hinblick auf das objektorientierte Programmieren und die Sprache C# gewonnen.

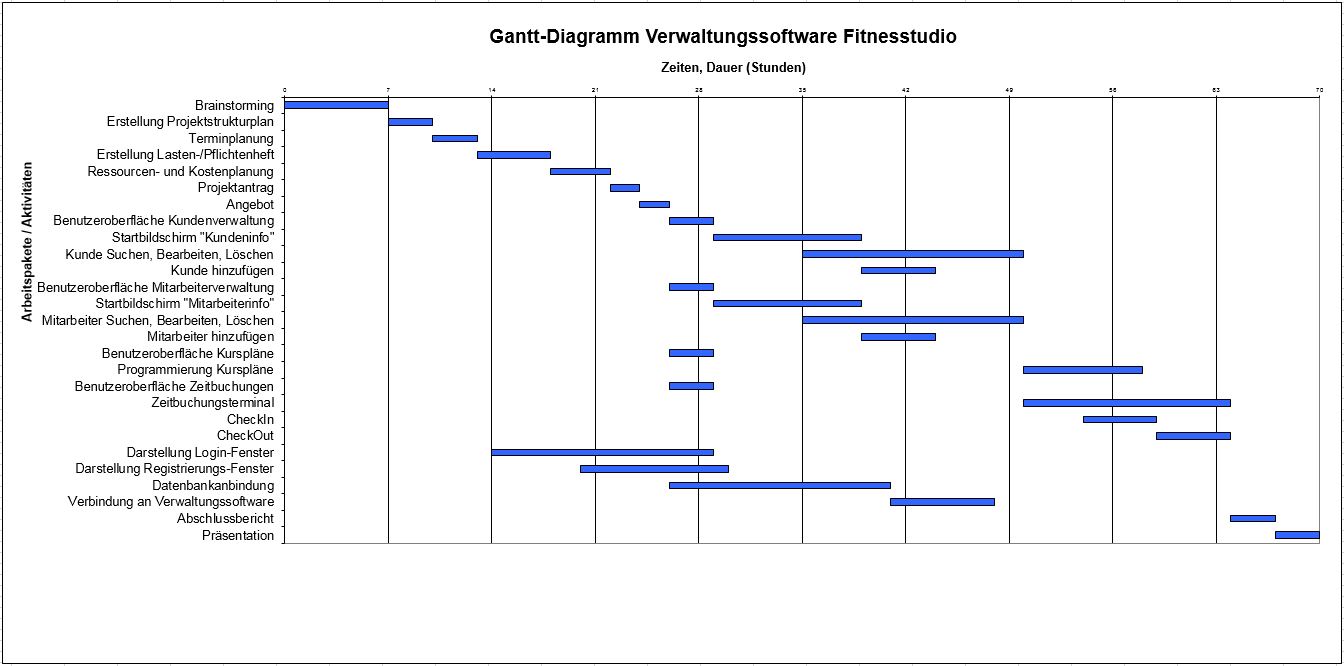
Für weitere Projekte kann die Gruppe die Erfahrung mitnehmen, rechtzeitig mit der Planung anzufangen und sich nicht zu lange an einer Teilaufgabe aufzuhalten. Abschließend kann somit gesagt werden, dass die Entwicklung des Verwaltungsprogramms nicht nur für das Fitnessstudio, sondern auch für das Team eine Bereicherung war, da dieses erste Projekterfahrungen sammeln konnte.

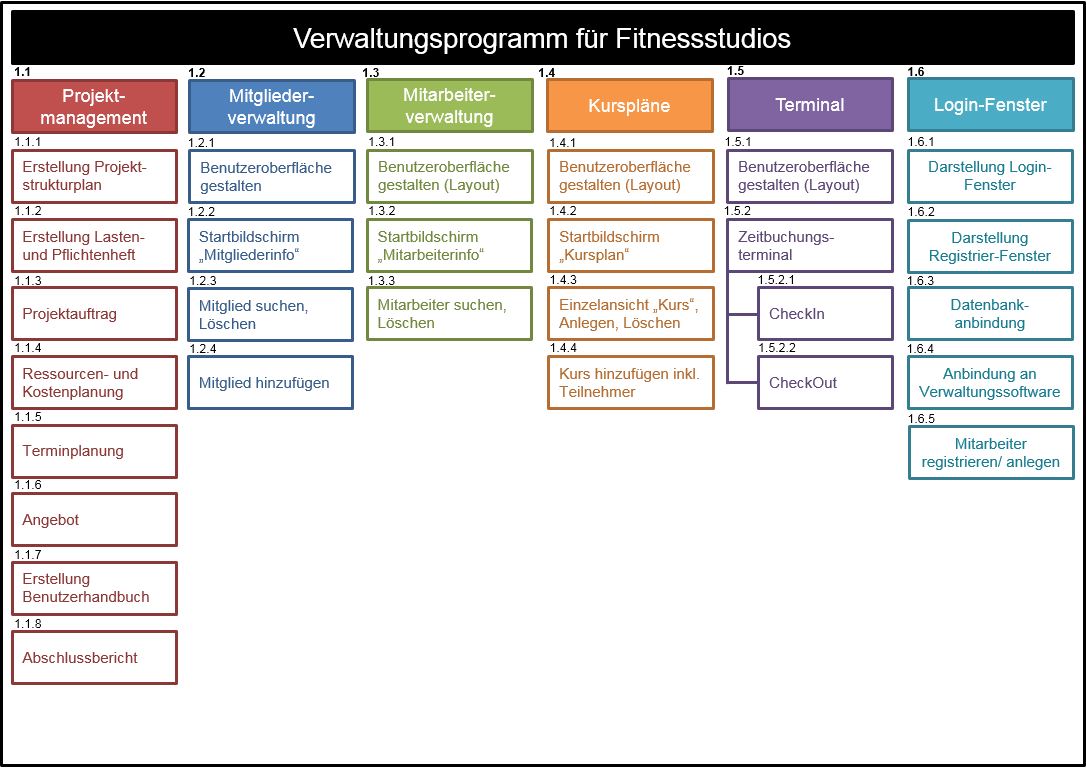
* 1. **Ausblick**

Obwohl alle im Lastenheft festgelegten Anforderungen erfüllt wurden, können in Zukunft neue Erweiterungsvorschläge des Kunden entwickelt werden. Beispielsweise wäre dies ein Terminkalender, in dem alle Beratungstermine zwischen Mitgliedern und Mitarbeitern oder die Urlaubsplanung der Mitarbeiter eingetragen werden können. Außerdem wäre es möglich, falls das Fitnessstudio irgendwann weitere Tochterstudios betreibt, die Software dort ebenfalls einzuführen.

**Anhang**

**A.1 Gantt-Diagramm**

****

**A.2 Projektstrukturplan**

**A.3 Verwendete Ressourcen**

**Hardware:**

* 4x Büroarbeitsplatz mit Notebook

**Software:**

* Windows 7/10 Betriebssystem
* Visual Studio Professional 2013 – Entwicklungsumgebung C#
* MySQL Datenbanksystem
* XAMPP – Verwaltungswerkzeug für MySQL Datenbank
* Git – Verteilte Versionsverwaltung

**Personal:**

* Inhaber und Mitarbeiter von FIT – Festlegung der Anforderungen und Abnahme des Projektes
* 4 Entwickler – Umsetzung des Projekts

**A.4 Lastenheft**

1. **Ausgangssituation**

Die Projektidee entstand aufgrund der aktuellen Mitgliederverwaltung. Bisher werden alle mitgliederbezogenen Daten noch in Papierform verwaltet, was sehr viel unnötige Zeit für die Trainer unseres Fitnessstudios in Anspruch nimmt. Ihre Hauptaufgabe sollte eigentlich in der Mitgliederbetreuung und in der Kursgebung liegen. Außerdem fehlt die Möglichkeit, Zeitbuchungen vorzunehmen, um zu sehen welche Mitglieder aktuell im Studio trainieren.

Da die Digitalisierung auch die Branche Fitness schon lange erreicht hat, besteht hier unbedingt Handlungsbedarf, um den Schritt der Digitalisierung nicht zu verpassen.

1. **Zielsetzung**

* Am Ende des Projektes soll eine übersichtliche Verwaltungssoftware entstanden sein, die eine leichte Bedienung für die Mitarbeiter des Fitnessstudios ermöglicht.
* Zur Überprüfung der Umsetzung soll es jede Woche ein Meeting mit den Mitarbeitern des Fitnessstudios und den Entwicklern geben, wo die leichte Bedienbarkeit in jedem Projektschritt sofort überprüft wird. Damit sollen spätere Änderungen aufgrund anderer Vorstellungen verhindert werden.
* In jedem Meeting soll ein weiterer Meilenstein erreicht worden sein. Die Fertigstellung des kompletten Projektes soll bis spätestens 11.07.2018 vollendet sein.

1. **Produkteinsatz**

* Das Produkt wird nur im Fitnessstudio XYZ eingesetzt und soll auch entsprechend nach diesen Anforderungen entwickelt werden.
* Verwendet wird das Produkt von den Mitarbeitern des Studios, aber nicht von den Mitgliedern.

1. **Funktionale Anforderungen**

* Kundendatenverwaltung:

Alle Kundendaten/Mitgliederdaten sollen mithilfe der Software verwaltet werden. Dazu gehören Kontaktdaten, Zeitbuchungen, ein Foto und die Verträge, Kurse und Scans (Personalausweis etc.). Es soll zum einen die Möglichkeit geben, nach einem Mitglied zu suchen und die Daten anschließend zu bearbeiten oder das Mitglied zu löschen. Zum anderen soll man ein neues Mitglied hinzufügen können.

* Mitarbeiterdatenverwaltung:

Alle Mitarbeiterdaten sollen mithilfe der Software verwaltet werden. Dazu gehören Kontaktdaten, Zeitbuchungen, ein Foto, sowie die Arbeitsverträge, Kurse (die der Mitarbeiter gibt) und Scans. Es soll zum einen die Möglichkeit geben, nach einem Mitglied zu suchen und die Daten anschließend zu bearbeiten oder den Mitarbeiter zu löschen. Zum anderen soll man einen neuen Mitarbeiter hinzufügen können.

* Kurspläne:

Es soll eine Übersicht über alle Kurse pro Woche entstehen. In der Detailansicht eines Kurses soll mindestens der Trainer, der Raum und die Anzahl der Teilnehmer zu sehen sein, die Namen der Teilnehmer hingegen nur, wenn dies möglich ist. In dieser Ansicht soll der Kurs geändert oder gelöscht werden können. Eine weitere Funktion soll das Hinzufügen eines neuen Kurses sein. Auch wieder optional wäre hier das gleichzeitige Hinzufügen der teilnehmenden Mitglieder.

* Zeitbuchungen:

Wenn ein Kunde zum Trainieren kommt, dann sollte der Mitarbeiter das Mitglied über die Eingabe der Nummer und einen Button „einchecken“ können. Durch das Einchecken wird er in einer Liste angezeigt, wo alle derzeit trainierenden Mitglieder angezeigt werden. Diese Liste soll als Übersicht für die Mitarbeiter dienen, aber die Daten auch gespeichert werden, damit man sehen kann wann und wie lange welches Mitglied zum Trainieren kam. Ist der Kunde fertig mit trainieren, soll er in der Liste ausgewählt und ausgecheckt werden können.

1. **Nichtfunktionale Anforderungen**

* Erweiterbarkeit:

Das Programm soll so entwickelt werden, dass später eventuell noch ein Terminkalender hinzuprogrammiert werden kann. Zum jetzigen Zeitpunkt wird dieser aber nicht gebraucht.

* Design:

Das Design soll an die Corporate Design Vorschriften des Fitnessstudios XYZ angepasst sein.

* Zuverlässigkeit:

Das Programm muss ausfallsicher sein, da es täglich gebraucht wird und wichtige, personenbezogene Daten enthält.

* Bedienbarkeit:

Das Programm soll sehr einfach zu bedienen sein und keine komplexen Funktionen enthalten.

1. **Lieferumfang**

* Verwaltungssoftware
* Desktop PC

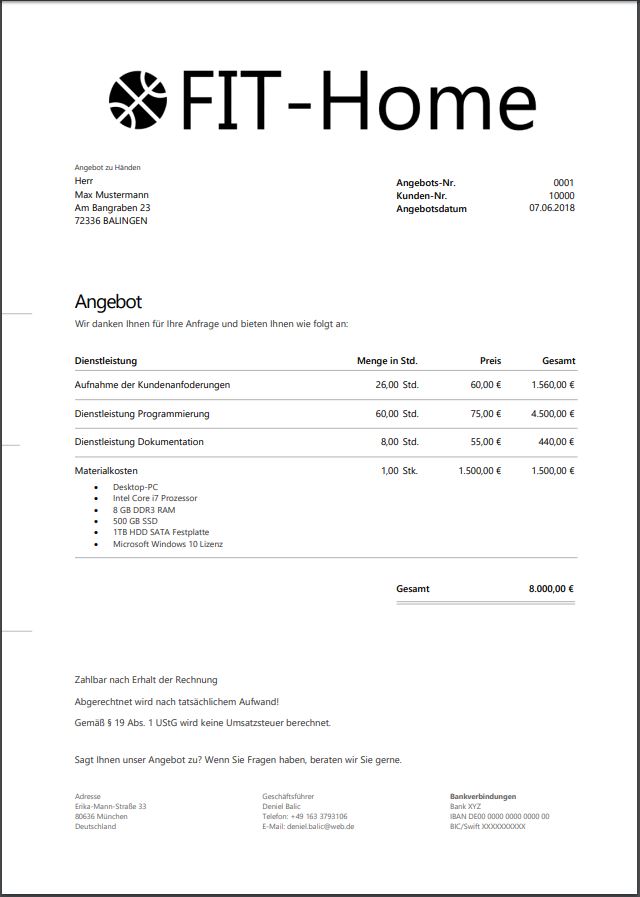
1. **Projektphasen und Meilensteine**

* Projektphasen:
  + Initialisierungs- und Definitionsphase
  + Planungsphase
  + Realisierungsphase
  + Abschlussphase
* Meilensteine:
  + Projektdefinition steht
  + Projektantrag freigegeben
  + Projektstart erfolgt
  + Planungen abgeschlossen
  + Umsetzung abgeschlossen
  + Projektabschluss

1. **Abnahmekriterien und Qualitätsanforderungen**

* Einfach zu bedienendes und ausfallsicheres Verwaltungsprogramm
* Berichte der Projektleitung werden vor dem Inhaber und den Mitarbeitern des Fitnessstudios vorgetragen
* Qualitätsanforderungen:

**A.5 Angebot und Projektauftrag**

****

|  |  |
| --- | --- |
| Projektname: Verwaltungsprogramm für Fitnessstudios | |
| Projektstart (Planung): 16.05.2018 | Starttermin: 07.06.2018 |
| Realisierungsereignis: Anforderung | Termin: 12.07.2018 |
| Projektabschluss  (Präsentation/Feedback: 12.07.2018 | Endtermin: 12.07.2018 |
| Vorprojektphase (Was muss vor dem Projekt geklärt/erledigt sein?):   * Entwicklungsumgebung festlegen * Design erstellen * Aufgabenverteilung | Nachprojektphase (Was muss/soll/kann nach dem Projekt umgesetzt werden?):   * Nachkalkulation * Projekterfahrungen festhalten und evtl. für andere Projektteams zur Verfügung stellen |
| Projektziele:   * Mitgliederverwaltung * Mitarbeiterdatenverwaltung * Zeitbuchungssystem * Kursplaner * Registrierung und Login | Nicht-Ziele:   * Statistiken erstellen * Terminplaner * Mitarbeitereinsatzplanung * Trainingsplanerstellung |
| Projektphasen/Hauptaufgaben It. PSP:   * Planung * Programmierung * Test | Betrachtungsobjekte (Übersicht):   * Benutzeroberfläche * Datenbankanbindung |
| Zusammenhang zu Strategien der Organisation und Beziehungen zu anderen Projekten:   * evtl. Folgeprojekt mit Umsetzung eines Terminkalenders | |
| Kosten/Ressourcen:   * Siehe Kosten- und Ressourcenplanung | |
| Projektteammitglieder/-mitarbeiter:   * Deniel Balic * Jens Dreher * Laura Knupfer * Jonas Reinecke | |
| Projektauftraggeber/-in  Datum, Name | Projektleiter/-in |

**A.6 Pflichtenheft**

1. **Ausgangssituation**

Das Fitnessstudio XYZ kam auf Grund ihrer aktuellen Mitgliederverwaltung auf unser Entwicklerteam zu. Bisher werden alle mitgliederbezogenen Daten noch in Papierform verwaltet, was sehr viel unnötige Zeit für die Trainer des Fitnessstudios in Anspruch nimmt. Ihre Hauptaufgabe sollte eigentlich in der Mitgliederbetreuung und in der Kursgebung liegen. Außerdem fehlt die Möglichkeit, Zeitbuchungen vorzunehmen, um zu sehen welche Mitglieder sich aktuell im Studio befinden. Dies kann zu Unstimmigkeiten gegenüber den geschlossenen Verträgen führen.

Da die Digitalisierung auch die Branche Fitness schon lange erreicht hat, besteht hier unbedingt Handlungsbedarf, um den Schritt der Digitalisierung nicht zu verpassen.

1. **Zielsetzung**

* Am Ende des Projektes soll eine übersichtliche Verwaltungssoftware vorhanden sein, die eine leichte Bedienung für die Mitarbeiter des Fitnessstudios ermöglicht.
* Zur Überprüfung der Umsetzung soll es jede Woche ein Meeting mit den Mitarbeitern des Fitnessstudios und unseren Entwicklern geben, wo die leichte Bedienbarkeit in jedem Projektschritt sofort überprüft wird. Damit sollen spätere Änderungen aufgrund anderer Vorstellungen verhindert werden.
* In jedem Meeting stellen unsere Entwickler einen weiterer Meilenstein vor, der erreicht worden ist. Die Fertigstellung des kompletten Projektes soll bis spätestens 11.07.2018 vollendet sein.

1. **Produkteinsatz**

* Das Produkt wird nur im Fitnessstudio XYZ eingesetzt und soll auch entsprechend genau nach diesen Anforderungen entwickelt werden.
* Verwendet wird das Produkt von den Mitarbeitern des Studios, aber nicht von den Mitgliedern.

1. **Funktionale Anforderungen**

* Kundendatenverwaltung:

Alle Kundendaten/Mitgliederdaten sollen mithilfe der Software verwaltet werden. Dazu gehören Kontaktdaten, Zeitbuchungen, ein Foto und die Verträge, Kurse und Scans (Personalausweis etc.). Es soll zum einen die Möglichkeit geben, nach einem Mitglied zu suchen und die Daten anschließend zu bearbeiten oder das Mitglied zu löschen. Zum anderen soll man ein neues Mitglied hinzufügen können.

* Mitarbeiterdatenverwaltung:

Alle Mitarbeiterdaten sollen mithilfe der Software verwaltet werden. Dazu gehören Kontaktdaten, Zeitbuchungen, ein Foto, sowie die Arbeitsverträge, Kurse (die der Mitarbeiter gibt) und Scans. Es soll zum einen die Möglichkeit geben, nach einem Mitglied zu suchen und die Daten anschließend zu bearbeiten oder das Mitglied zu löschen. Zum anderen soll man einen neuen Mitarbeiter hinzufügen können.

* Kurspläne:

Es soll eine Übersicht über alle Kurse pro Woche entstehen. In der Detailansicht eines Kurses soll mindestens der Trainer, der Raum und die Anzahl der Teilnehmer zu sehen sein, die Namen der Teilnehmer hingegen nur, wenn dies möglich ist. In dieser Ansicht soll der Kurs geändert oder gelöscht werden können. Eine weitere Funktion soll das Hinzufügen eines neuen Kurses sein. Auch wieder optional wäre hier das gleichzeitige Hinzufügen der teilnehmenden Mitglieder.

* Zeitbuchungen:

Wenn ein Kunde zum Trainieren kommt, dann sollte der Mitarbeiter das Mitglied über die Eingabe der Nummer und einen Button „einchecken“ können. Durch das Einchecken wird er in einer Liste angezeigt, wo alle derzeit trainierenden Mitglieder angezeigt werden. Diese Liste soll als Übersicht für die Mitarbeiter dienen, aber die Daten auch gespeichert werden, damit man sehen kann wann und wie lange welches Mitglied zum Trainieren kam. Ist der Kunde fertig mit trainieren, soll er in der Liste ausgewählt und ausgecheckt werden können.

1. **Nichtfunktionale Anforderungen**

* Erweiterbarkeit:

Das Programm soll so entwickelt werden, dass später eventuell noch ein Terminkalender hinzuprogrammiert werden kann. Zum jetzigen Zeitpunkt wird dieser aber nicht gebraucht.

* Design:

Das Design soll an die Corporate Design Vorschriften des Fitnessstudios XYZ angepasst sein.

* Zuverlässigkeit:

Das Programm muss ausfallsicher sein, da es täglich gebraucht wird und wichtige, personenbezogene Daten enthält.

* Bedienbarkeit:

Das Programm soll sehr einfach zu bedienen sein und keine komplexen Funktionen enthalten.

1. **Lieferumfang**

* Verwaltungssoftware
* Desktop PC

1. **Projektphasen und Meilensteine**

* Projektphasen:
  + Initialisierungs- und Definitionsphase
  + Planungsphase
  + Realisierungsphase
  + Abschlussphase
* Meilensteine:
  + Projektdefinition steht
  + Projektantrag freigegeben
  + Projektstart erfolgt
  + Planungen abgeschlossen
  + Umsetzung abgeschlossen
  + Projektabschluss

1. **Abnahmekriterien und Qualitätsanforderungen**

* Einfach zu bedienendes und ausfallsicheres Verwaltungsprogramm
* Berichte der Projektleitung werden vor dem Inhaber und den Mitarbeitern des Fitnessstudios vorgetragen
* Qualitätsanforderungen:
* Leichte und Verständliche Bedienung, sodass durch die Schulung der Mitarbeiter kein großer Aufwand entsteht
* Hohe Ausfallsicherheit, da die Mitarbeiter bei Ausfällen keinen Überblick über die teilweise begrenzte Trainingsdauer der Mitglieder haben
* Hohe Geschwindigkeit des Programms, weil es laut den Mitarbeitern oft Stoßzeiten gibt an welchen viele Mitglieder einchecken und es nicht zu Wartezeiten kommen soll

**A.7 Aufgabenverteilung**

|  |  |
| --- | --- |
| **Aufgabenabschnitt** | **Verantwortliche/r** |
| Projektmanagement | **Laura Knupfer** |
| Mitgliederverwaltung | **Jonas Reinecke** |
| Mitarbeiterverwaltung | **Deniel Balic / Jonas Reinecke** |
| Terminal | **Deniel Balic** |
| Kursplan | **Jens Dreher / Laura Knupfer** |
| Registrierung und Login | **Jens Dreher / Laura Knupfer** |

**A.8 Arbeitsprotokolle**

