Pflichtenheft – Sportstudio

1. Einleitung

Dieses Pflichtenheft beschreibt die Anforderungen und Spezifikationen für die Entwicklung einer Verwaltungsapplikation für ein Sportstudio. Das Ziel der App besteht darin, eine benutzerfreundliche Oberfläche anzubieten, auf der sich Benutzer als Admin oder regulärer Nutzer anmelden können. Als Admin haben sie die Möglichkeit, registrierte Benutzer, sowie deren Abonnement zu verwalten.

1. Projektbeschreibung

* Die Anwendung ermöglicht es Benutzern, sich als Admin oder regulärer Nutzer anzumelden.
* Funktionen umfassen Abo-Buchung, Anzeige von unseren Angeboten und deren Infos, Registrierung, An- und Abmeldung und für Admins die Möglichkeit, Daten anderer User zu sehen und zu verwalten.
* Die App besteht aus verschiedenen Ansichten, darunter die Startseite, "Abo ", "Anmeldung" und "Datenansicht" für Admins.

1. Anforderungen

3.1. Benutzeranmeldung und -verwaltung

* Benutzer registrieren sich mit Vor- und Nachnamen, E-Mail-Adresse, Passwort sowie deren Adresse. Die Auswahl zur Admin-Rolle ist hier nicht gegeben. Das heißt, es ist nicht möglich sich als Admin zu registrieren, da dies direkt im Code von den Entwicklern je nach Beauftragung gemacht wird.

3.2. Abo-Buchung

* Nutzer können verschiedene Abo-Optionen anzeigen und auswählen.
* Die Auswahl erfolgt durch Klicken auf die entsprechenden Buttons, die die verschiedenen Abo-Optionen repräsentieren.
* Jede ausgewählte Option erhöht den insgesamten monatlichen Preis des Abonnements.
* Der „Abo-Bundle“ bietet die Funktion, alle bestehenden Abo-Optionen auszuwählen, worauf der User anschließend einen Rabatt erhält, da er sich für alle Optionen entschieden hat.
* Nach Auswahl können Nutzer die Buchung bestätigen und erhalten eine Bestätigungsmeldung.

3.3. Raumansicht

* Benutzer können verfügbare Räume ansehen.
* Die Ansicht zeigt Details zu jedem Raum, wie Größe, Ausstattung, die zu trainierenden Muskel/Muskelgruppen und designierte Coaches.

3.4. Registrierung und An-/Abmeldung

* Neue Benutzer müssen sich registrieren und persönliche Daten eingeben.
* Benutzer können sich an- und abmelden, wobei ihre Anwesenheit im System aktualisiert wird.

3.5. Benutzerverwaltung für Admins

* Admins können registrierte Benutzer anzeigen und verwalten.
* Diese Ansicht ermöglicht es Admins, Benutzerdetails einzusehen, zu bearbeiten und Benutzer zu löschen.

3.6. Absicherung

* Nutzer können sich nur registrieren, wenn alle eingegebenen Daten den vorgegebenen Regeln entsprechen.
* Die Anmeldung erfolgt nur dann, wenn die eingegebenen Daten mit den abgespeicherten Daten des Nutzers übereinstimmen.

1. Design und Benutzeroberfläche

* Eine einfache und intuitive Benutzeroberfläche ermöglicht ein müheloses Anmelden und Buchen eines Abonnements, sowie eine gute Übersicht der Räume im Sportstudio.
* Die Darstellung der Informationen wie die Auswahl der verschiedenen Abonnement-Möglichkeiten und deren Preise ist klar und übersichtlich gestaltet.

1. Technische Anforderungen

* Plattform: Desktop-Anwendung für Windows
* Entwicklungssprache und Frameworks: C# mit dem .NET-Framework.

1. Projektentwicklung

Die Entwicklung der SportStudio-App wird in mehreren Phasen durchgeführt:

6.1. Phase: Anforderungsanalyse und Planung

* In dieser Phase werden die Anforderungen an die App detailliert analysiert und das Pflichtenheft erstellt.
* Das Entwicklungsteam führt Besprechungen mit dem Auftraggeber (Lehrer) durch, um alle Anforderungen zu klären und zu verstehen.
* Am Ende dieser Phase wird ein detaillierter Projektplan erstellt, der Vorgänge, Ziele und Aufgaben umfasst.

6.2. Phase: Design und Benutzeroberfläche

* In dieser Phase werden das Design und die Benutzeroberfläche entwickelt.
* Das Entwicklungsteam erstellt Skizzen, um das visuelle Konzept zu veranschaulichen und Feedback einzuholen.

6.3. Phase: Entwicklung

* In dieser Phase werden das Design und die Benutzeroberfläche entwickelt.
* Das Entwicklungsteam arbeitet gemäß den Anforderungen und dem Design, um die Funktionalitäten der App umzusetzen.
* Regelmäßige Meetings und Code-Reviews werden durchgeführt, um die Qualität des Codes sicherzustellen und Fehler frühzeitig zu erkennen und zu beheben.

6.4. Phase: Test und Qualitätssicherung

* In dieser Phase werden umfangreiche Tests durchgeführt, um sicherzustellen, dass alle Funktionen der App einwandfrei funktionieren.

1. Herausforderungen und Probleme

Die Entwicklung einer Anwendung wie der SportStudio-App bring eine Reihe von Herausforderungen und potenziellen Problemen mit sich, die während des Entwicklungsprozesses auftreten können. In diesem Kapitel werden die wichtigsten Herausforderungen und Probleme identifiziert und beschrieben, die bei der Umsetzung der Anwendung aufgetreten sind. Ziel ist es, diese Herausforderungen frühzeitig zu erkennen und geeignete Lösungsansätze zu entwickeln, um die reibungslose Funktionalität, Sicherheit und Benutzerfreundlichkeit der Anwendung sicherzustellen.

7.1. Anpassung der App-Formen

* Beim Öffnen der Login-Seite und der Abonnement-Seite, kamen wir zur Konklusion, diese in einem neuen Fenster öffnen zu lassen. Der Grund dafür liegt in der besseren Übersichtlichkeit im Code und insbesondere in der „Entwurf“-Ansicht in Visual Studio, während der Entwicklung. Da ohne dies viele verschiedene Windows Forms Elemente wie „Labels“, „Text-Boxen“ und „Buttons“ übereinander liegen würden, was den Entwicklungsprozess um einiges erschwert hätte.

7.2. Effektive Nutzung von Variablen über Formen hinweg

* Ein Problem bestand darin, bestimmte Variablen über verschiedene Formen hinweg effektiv zu nutzen. Um die Datenkonsistenz zu gewährleisten, wurden diese Variablen als „public“ deklariert. Dies ermöglichte einen einfachen Austausch von Informationen zwischen den Formen.
* Zusätzlich wurden manche Felder mit „static“ deklariert, um sicherzugehen, dass in der gesamten Anwendung, dieses Feld nur eine einzige Instanz hat.

7.3. Input Felder und deren Kriterien

* Bei den Input Feldern zur Registrierung, traten Komplikationen auf, in Bezug auf das Blockieren von bestimmten Eingabewerten wie Buchstaben oder anderen Sonderzeichen. Dies wird im folgenden Kapitel „Code-Struktur und Implementierungsdetails“ deutlicher veranschaulicht.

1. Code-Struktur und Implementierungsdetails

In diesem Abschnitt werden die wichtigsten Aspekte der Code-Struktur und der Implementierungsdetails der SportStudio-App erläutert. Es werden zum großen Teil Methoden und andere Zusammenhänge beschrieben, die für die Funktionalität der App relevant sind.

8.1. Form1 (Home)

Die From1 ist das Herzstück des SportStudio-Projekts und enthält den Code für das Hauptformular "Home" einer Windows Forms-Anwendung.

* Konstruktor: Der Konstruktor "Form1" initialisiert die HomePage der Anwendung, und erstellt ebenso ein paar Benutzer und Admins, welche anschließend in eine statische (sprich eine einzigartige und global verfügbare) „users“ oder „admins“ Liste hinzugefügt werden.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Als Folge wird dieses Fenster geladen:

Ein Bild, das Text, Screenshot, Cartoon enthält.

Automatisch generierte Beschreibung

* Weiterer Code in dieser Klasse bezieht sich auf wesentliche „Click-Events“, welche ausgeführt werden, wenn ein User mit der Maus auf eines der gebenen Elemente klickt, wie zum Beispiel ein Bild zu den verschiedenen Räumen oder den Anmeldungslink.

8.2. Raumansicht

Ein Bild, das Text, Screenshot, Website, Onlinewerbung enthält.

Automatisch generierte Beschreibung

* Sobald ein User auf eines der Bilder zu den Räumen mit der Maus klickt, gelangt er auf die entsprechende Informationsansicht zu diesem Raum.
* Bezüglich des Aufbaus der Raumansicht, gibt es kaum selbst geschrieben Anpassungen im Code, da die zu sehende Elemente und Texte, direkt mit der Maus zur Entwicklungszeit versteckt oder angezeigt werden können.
* Ein Bild, das Text, Schrift, Screenshot, weiß enthält.

  Automatisch generierte BeschreibungSollte Mitglieder ein Abonnement haben für diesen Raum werden diese bewusst oben recht angzeigt.
* Mit dem Button oben rechts, gelangt man von der aktuell angezeigten Bildansicht zurück zur HomePage.

Ein Bild, das Screenshot, Stecker, Design enthält.

Automatisch generierte Beschreibung

8.3. Abo-Screen

Ein Bild, das Text, Screenshot, Schrift, Software enthält.

Automatisch generierte Beschreibung

* In dieser Ansicht findet ein User alle möglichen Angebote des SportStudios und kann so sein Abonnement für sich selbst anpassen, wie er möchte.
* Mit dem Anklicken jenes Moduls steigt der monatliche Preis des Abonnements.
* Es wird allerdings auch die Möglichkeit für ein sog. Abo-Bundle angeboten, welches durch Anklicken alle Module auswählt und anschließend dem User einen Rabatt bietet. In diesem Fall um 5 Euro.

8.4. Anmeldung und Registrierung

* Sobald sich das Anmeldungsfenster öffnet, ist einem die Möglichkeit gegeben sich direkt anzumelden.  
  Durch den Button rechts oben kann man zwischen der Anmeldung und der Registrierung wechseln.

Ein Bild, das Text, Screenshot, Software, Webseite enthält.

Automatisch generierte Beschreibung

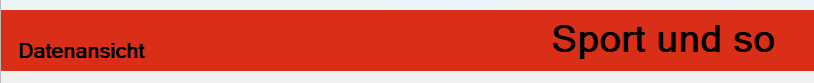
Ein Bild, das Text, Screenshot, Software, Zahl enthält.

Automatisch generierte Beschreibung

* Da diese Klasse einen neuen Dialog bzw. ein neues Fenster in der Anwendung öffnet, wird an erster Stelle in dessen Code eine Konstruktor Methode benötigt. Mithilfe von „InitializeComponent()“ erstellen wir anschließend alle sich darin befindenden Elemente des Formulars.

Ein Bild, das Text, Screenshot, Schrift, Design enthält.

Automatisch generierte Beschreibung

* Zusätzlich benötigen wir in dieser Klasse Zugriff auf die Instanz der HomePage. Der Grund dafür ist, dass wir Anhand des angemeldeten Users (ob Admin oder regulär) bestimmte Links/Buttons freischalten. Im Falle eines Admin Users, wird in der Navigationszeile der HomePage folgender Link freigeschaltet.
* Sobald man sich angemeldet hat, wird man durch eine MessageBox begrüßt, es steht einem nur noch der AbmeldeButton zu Verfügung und der Name des aktuellen Nutzers wird angezeigt.

8.5. Datenverwaltung

Ein Bild, das Text, Reihe, Zahl, Schrift enthält.

Automatisch generierte Beschreibung

* In diesem neuen Fenster kann ein angemeldeter Admin alle bestehenden User anschauen und verwalten. Hier können die Felder direkt bearbeitet werden.
* Die erste Spalte veranschaulicht die von C# automatisch generierten Ids.

8.6. Persistenz oder Session-basiert

Wir haben uns bewusst dazu entschieden, auf Persistenz zu verzichten und stattdessen alles auf die Session zu stützen. Diese Entscheidung mag auf den ersten Blick ungewöhnlich erscheinen, aber sie war für dieses Projekt durchaus angemessen.

1. Fazit

Die SportStudio-App wurde sorgfältig entwickelt, beginnend mit klaren Anforderungen und einem gut strukturierten Plan. Das Ziel war eine einfache, benutzerfreundliche Anwendung für die Verwaltung von Abonnements und Räumen in einem Sportstudio.

Das Design der App wurde übersichtlich gestaltet, um eine einfache Navigation zu ermöglichen. Technisch wurde die App für Windows-Desktops mit C# und dem .NET-Framework entwickelt.

Während der Entwicklung wurden einige Probleme wie die Anpassung von Formen und die Nutzung von Variablen gelöst. Trotzdem konnte die App erfolgreich umgesetzt werden und wurde gründlich getestet, um sicherzustellen, dass sie reibungslos funktioniert.

Insgesamt bietet die SportStudio-App eine praktische Lösung für die Verwaltung von Abonnements und Räumen in einem Sportstudio.

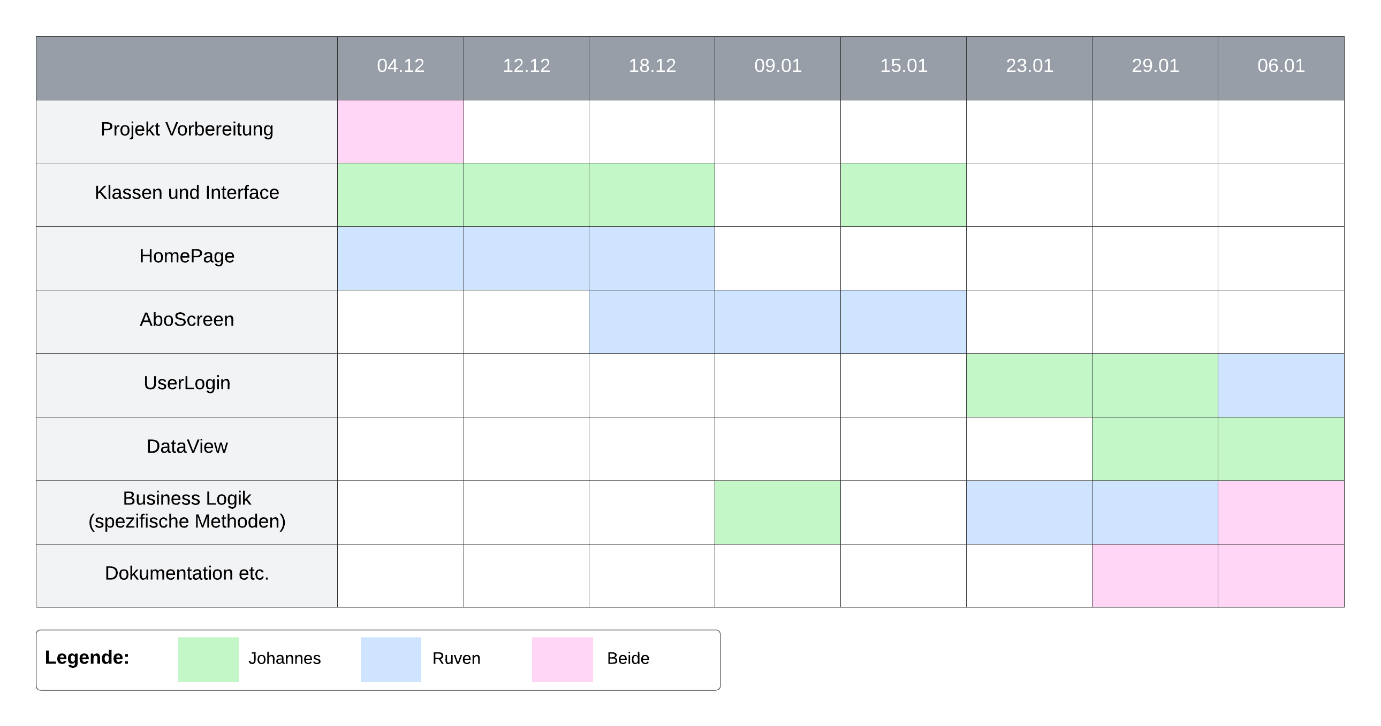
Alles in allem kam es zu einem guten Lernprozess, einem erfolgreichen Projekt und etwas Spaß.

1. Struktogramm

Ein Bild, das Text, Screenshot, Reihe, Schrift enthält.

Automatisch generierte Beschreibung

1. Gantt-Diagramm



1. UML-Klassendiagramm

Ein Bild, das Text, Diagramm, Screenshot, Design enthält.

Automatisch generierte Beschreibung

1. Use-Case-Diagramm

Ein Bild, das Text, Diagramm, Reihe, Screenshot enthält.

Automatisch generierte Beschreibung

1. Sequenzdiagramm

Ein Bild, das Text, Diagramm, Reihe, Screenshot enthält.

Automatisch generierte Beschreibung