

Projektaufgabe INF2015: C# und .NET - „Liga Manager“

Das aufstrebende Jungunternehmen „11 Kumpels UG (haftungsbeschränkt)“ beauftragt Sie mit der Entwicklung des Prototyps einer Software zur Abbildung des Geschäftsmodells.

„11 Kumpels UG (haftungsbeschränkt)“ möchte eine Anwendung zum einfachen Durchführen eines Bundesligatippspiels anbieten und mit diesem höchstinnovativen Konzept ein führendes Startup in einem aufstrebenden Markt werden.

In einem vorherigen Projekt wurde ein Prototyp bei einem anderen Dienstleister in Auftrag gegeben. Nach erfolgreicher Erstellung der Datenbank konnte dieser Dienstleister allerdings nicht den Vertrag erfüllen. Aus diesem Projekt existiert bereits die Datenbank (SQLite-Format), die auch die Basis für den neuen Prototyp werden soll.

Zugang zum Prototyp aus dem vorherigen Projekt kann aus lizenzrechtlichen Gesichtspunkten nicht für die Neuentwicklung angeboten werden. Zugang zur Datenbank ist selbstverständlich möglich.

Lesen Sie zuerst die allgemeinen Tipps zur Lösung der Projektaufgabe im Anhang!

Aufgabenstellung (100 Punkte)

Entwickeln Sie für das Unternehmen ein modernes System zur Durchführung eines Bundesligatippspiels, welches kompatibel zur bestehenden Datenbank ist.

Die Realisierung muss mit C#, WPF und WCF auf Basis von .NET 4.5 oder 4.6 erfolgen. Die Verwendung von MVVMC und Fluent NHibernate mit Repository-Pattern wird vorgeschrieben.

Die Software soll als Client-Server-Software mittels Verwendung von WCF (wsHttpBinding) realisiert werden. MVVM-Frameworks können nicht verwendet werden.

Es werden nachfolgend nur die zu realisierenden Funktionalitäten beschrieben. Bei der eigentlichen Umsetzung werden den Studierenden große Freiheiten eingeräumt. Dies soll kreative Eigenleistungen fördern und deren Bewertung ermöglichen. Dies bezieht sich sowohl auf die Architektur der Anwendung als auch auf das Design der Oberflächen und das Bedienkonzept der Anwendung.

Die bestehende Datenbank soll zur Entwicklung der neuen Software analysiert und genutzt werden. Hierfür stehen im Internet verschiedene Open-Source-Programme zur Verwaltung von SQLite-Datenbanken zur Verfügung.

Das vorliegende Datenbankschema darf nicht verändert werden.

Auftretende Programmabstürze führen zu Punktabzug!

Fehlende Eingabevalidierungen führen ebenfalls zu Punktabzug, auch wenn das Programm nicht abstürzt.
Lässt sich das abgegebene Projekt nicht fehlerfrei kompilieren und starten, führt dies ebenfalls zu Punktabzug!

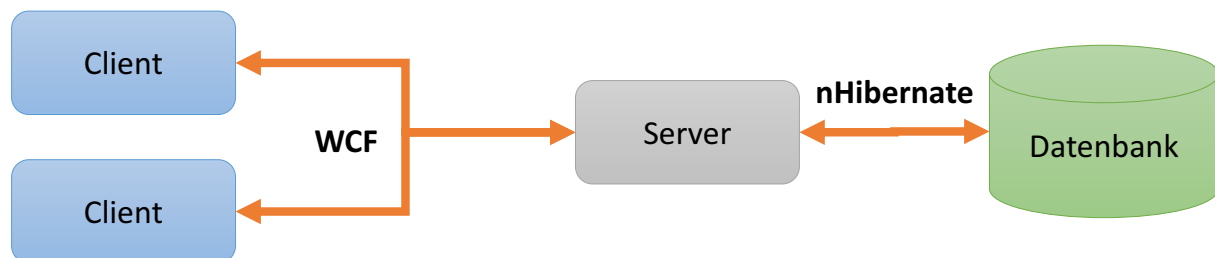
Fehlermeldungen müssen jederzeit klar formuliert, für den Anwender verständlich und situationsbezogen sein. Ansonsten führt dies zu Punktabzug!

Von den Studierenden wird die Fähigkeit zur Recherche im Internet mittels geeigneter Suchmaschinen erwartet.

Hinweis: Das Punktesystem ist analog zur Bewertung von Studienarbeiten

Client-Server-System (10 Punkte):

Nachfolgend wird der grundlegende Aufbau beschrieben, den Sie umsetzen sollen.



Server:

Realisieren Sie den Server auf Basis einer Konsolenanwendung, die einen WCF-Dienst zur Kommunikation bereitstellt.

Implementieren Sie die Kommunikation mit der Datenbank mittels Fluent-nHibernate. Setzen Sie hierzu das Konzept des „Optimistic Locking“ um. Ermöglichen Sie den gleichzeitigen Zugriff von mehr als einer Client-Anwendung zur selben Zeit.

Verwaltungs-Client:

Realisieren Sie den Verwaltungs-Client auf Basis einer WPF-Anwendung, die Sie mit dem MVVMC-Entwurfsmuster aufbauen.

Der Client darf nicht direkt mit der Datenbank kommunizieren. Der Client darf ausschließlich mittels WCF mit dem Server kommunizieren.

Navigationsmöglichkeit:

Das Softwaresystem muss über eine Navigationsmöglichkeit zum Öffnen der nachgehend beschriebenen Module verfügen.
Realisieren Sie hierzu eine Menüführung, die die Auswahl der Module ermöglicht.

Saisonverwaltung:

Diese Funktionalität soll die Verwaltung von Saisons (z.B. 1. Bundesliga – Saison 2017/2018) ermöglichen.

Enthalten sollen Möglichkeiten zum Erstellen, Bearbeiten und Löschen von Saisons sein. Die einzelnen Saisons sollen in einer Listenansicht aufgeführt werden und in einer Detailansicht bearbeitet werden können.

Mannschaftsverwaltung:

Diese Funktionalität soll die Verwaltung von Mannschaften (z.B. FC Schalke 04) ermöglichen.

Enthalten sollen Möglichkeiten zum Erstellen, Bearbeiten und Löschen von Mannschaften sein.

Die einzelnen Mannschaften sollen in einer Listenansicht aufgeführt werden und in einer Detailansicht bearbeitet werden können.

Die Mannschaften sollen über eine Listenansicht in einem eigenen Detailbereich einer oder mehreren Saisons zugeordnet werden.

Spielverwaltung:

Diese Funktionalität soll die Verwaltung von Spielen ermöglichen. Spiele zwischen den Mannschaften sollen erstellt, bearbeitet und gelöscht werden können. Spiele werden in Spieltagen organisiert. Alle Spieltage einer Saison sollen automatisch generiert werden können. Der Import von Spieltagen soll möglich sein.

Die Spiele sollen über eine Listenansicht aufgeführt werden und in einer Detailansicht bearbeitet werden können. Die Listenansicht soll nach Spieltag gefiltert werden können.

Tipper-Verwaltung:

Diese Funktionalität soll die Verwaltung von Tippern (Spielern) ermöglichen. Tipper sollen erstellt, bearbeitet und gelöscht werden können.

Die Tipper sollen über eine Listenansicht aufgeführt werden und in einer Detailansicht bearbeitet werden können.

Tippspiel-Client:

Realisieren Sie den Tipp-Client auf Basis einer WPF-Anwendung, die Sie mit dem MVVMC-Entwurfsmuster aufbauen.

Der Client darf nicht direkt mit der Datenbank kommunizieren. Der Client darf ausschließlich mittels WCF mit dem Server kommunizieren.

Navigationsmöglichkeit:

Das Softwaresystem muss über eine Navigationsmöglichkeit zum Öffnen der nachgehend beschriebenen Module verfügen.

Realisieren Sie hierzu eine Menüführung, die die Auswahl der Module ermöglicht.

Tippverwaltung:

Diese Funktionalität soll das Tippen auf das Ergebnis von Spielen ermöglichen.

Enthalten sollen Möglichkeiten zum Erstellen, Bearbeiten und Löschen von Tipps sein. Die einzelnen Tipps sollen pro Spieltag für alle an diesen stattfindenden Spielen abgegeben werden können.

Hierzu sollen alle Spieltage einer Saison (nach Auswahl dieser) in einer Listenansicht aufgeführt werden. Das System soll die Abgabe, das Bearbeiten und das Löschen der Tipps in einem Detailbereich ermöglichen.

Saison-Tabelle:

Das Software-System soll die aktuelle Tabelle der Saison in einer Listenansicht darstellen können. Die Tabelle soll bis zum auswählbaren Spieltag berechnet werden.

Tipper-Rangliste:

Das Software-System soll eine Rangliste der Tipper pro Saison in einer Listenansicht darstellen können. Die Rangliste soll bis zum auswählbaren Spieltag berechnet werden.

Erscheinungsbild der Oberflächen:

Das Design soll sich an modernen Windows-Anwendungen orientieren und für Anwender ohne Expertenwissen intuitiv bedienbar sein.

Die Anwendung soll optisch ansprechend gestaltet werden. Hierbei ist auf die integrierten Funktionalitäten für Theming und Styling von WPF zurückzugreifen.

Um die ansprechende Gestaltung der Oberfläche zu unterstützen, finden Sie im Internet auch frei verfügbare Icon-Bibliotheken zur Gestaltung der Oberfläche (z.B. Schaltflächen, etc).

Verwaltung (50 Punkte)

Nachfolgend werden die Anforderungen an die Verwaltungsfunktionalität beschrieben. Sowohl die Client- als auch die Serverseite ist gemäß den Anforderungen umzusetzen.

Tipper-Verwaltung (5 Punkte):

Realisieren Sie folgende Funktionalitäten zur Abbildung der Verwaltung von Tippern. Möglichkeiten zur Anzeige, Neuanlage, zum Speichern, Bearbeiten und Löschen von Tippern.

Tipper müssen sich durch einen eindeutigen Spitznamen (ohne Beachtung von Groß- und Kleinschreibung, also wäre folgendes **nicht** möglich: „Horst“, „horst“) unterscheiden.

Ein Tipper darf nicht gelöscht werden, so lange es noch Tipps (Wetten) gibt, die ihm zugeordnet sind.

Dem Anwender soll dieser Sachverhalt in einer verständlichen Meldung (oder einem anderen benutzerfreundlichen Vorgehen) verdeutlicht werden.

Tipper zeichnen sich durch folgende Eigenschaften aus:

- Spitzname (Nickname | Spitzname (Anmeldename) des Tippers)
- Vorname (Firstname | Vorname des Tippers)
- Nachname (Lastname | Nachname des Tippers)

Vergessen Sie nicht die Abbildung des Optimistic Lockings!

Tabelle: Bettors

Die Tipper sollen in einer Listendarstellung angezeigt werden. Tipper sollen auswählbar sein. Für den ausgewählten Tipper soll eine Detailansicht eingeblendet werden, auf der die weiteren Eigenschaften verwaltet werden können.

Saison-Verwaltung (5 Punkte):

Realisieren Sie folgende Funktionalitäten zur Abbildung der Verwaltung von Saisons. Möglichkeiten zur Anzeige, Neuanlage, zum Speichern, Bearbeiten und Löschen von Saisons.

Saisons müssen sich durch einen eindeutigen Namen (ohne Beachtung von Groß- und Kleinschreibung, also wäre folgendes **nicht** möglich: „1. BL 16/17“, „1. bl 16/17“) unterscheiden.

Eine Saison darf nicht gelöscht werden, so lange es noch Mannschaften oder Spiele gibt, die ihr zugeordnet sind.

Dem Anwender soll dieser Sachverhalt in einer verständlichen Meldung (oder einem anderen benutzerfreundlichen Vorgehen) verdeutlicht werden.

Saisons zeichnen sich durch folgende Eigenschaften aus:

- Name (Name | Bezeichnung der Saison)
- Beschreibung (Description | optionaler beschreibender Text)
- Reihenfolge (Sequence | Sortierkriterium zur Steuerung der Anzeigereihenfolge)

Vergessen Sie nicht die Abbildung des Optimistic Lockings!

Tabelle: Seasons

Die Saisons sollen in einer Listendarstellung angezeigt werden. Saisons sollen auswählbar sein. Für die ausgewählte Saison soll eine Detailansicht eingeblendet werden, auf der die weiteren Eigenschaften verwaltet werden können.

Mannschaftsverwaltung (10 Punkte):

Realisieren Sie folgende Funktionalitäten zur Abbildung der Verwaltung von Mannschaften.

Schaffen Sie Möglichkeiten zur Anzeige, Neuanlage, zum Speichern, Bearbeiten und Löschen von Mannschaften.

Mannschaften zeichnen sich durch folgende Eigenschaften aus:

- Name (Name | Bezeichnung der Mannschaft)

Mannschaften müssen sich durch einen eindeutigen Namen (ohne Beachtung von Groß- und Kleinschreibung, also wäre folgendes **nicht** möglich: „SC Freiburg“, „Sc Freiburg“) unterscheiden.

Werden Mannschaften gelöscht, zu denen es noch Spiele gibt, sollen diese automatisch mitgelöscht werden.

Vergessen Sie nicht die Abbildung des Optimistic Lockings!

Tabelle: Teams, SeasonToTeamRelation

Die Mannschaften sollen in einer Listendarstellung angezeigt werden. Mannschaften sollen auswählbar sein. Für die ausgewählte Mannschaft soll eine Detailansicht eingeblendet werden, auf der die weiteren Eigenschaften verwaltet werden können.

Die Verknüpfung zu Saisons soll über eine eigene Listenansicht im Detailbereich der Mannschaften realisiert werden.

Zum Hinzufügen der Verknüpfung sollen die vorhandenen Saisons zur Auswahl angezeigt werden. Es darf pro Mannschaft nur eine Verknüpfung zur selben Saison angelegt werden. Dies ist möglichst anwenderfreundlich zu realisieren. Hinzugefügte Verknüpfungen zu Saisons sollen auch wieder entfernt werden können.

Spiel-Verwaltung (15 Punkte):

Realisieren Sie folgende Funktionalitäten zur Abbildung der Verwaltung von Spielen zwischen den Mannschaften.

Schaffen Sie Möglichkeiten zur Anzeige, Neuanlage, zum Speichern, Bearbeiten und Löschen von Spielen.

Spiele zeichnen sich durch folgende Eigenschaften aus:

- Spieltag (MatchDay | Spieltag der verknüpften Saison)
- Beginn (DateTime | Datum und Uhrzeit des Beginns des Spiels)
- Heim-Tore (HomeTeamScore | Tore der Heim-Mannschaft)
- Auswärts-Tore (AwayTeamScore | Tore der Auswärts-Mannschaft)
- Heim-Mannschaft (HomeTeamId | Verknüpfung zur Heim-Mannschaft)
- Auswärts-Mannschaft (AwayTeamId | Verknüpfung zur Auswärts-Mannschaft)
- Saison (SeasonId | Verknüpfung zur Saison)

Mannschaften können pro Saison nur 1 Mal gegeneinander spielen (bei gleicher Kombination von Heim- und Auswärtsmannschaft).

Folgendes ist also möglich: „FC Bayern München – Borussia Dortmund“ und „Borussia Dortmund – FC Bayern München“

Folgendes ist **nicht** möglich: „FC Bayern München – Borussia Dortmund“ und „FC Bayern München – Borussia Dortmund“.

Vergessen Sie nicht die Abbildung des Optimistic Lockings!

Tabelle: Matches

Die Spiele sollen in einer Listendarstellung angezeigt werden. Spiele sollen auswählbar sein. Für das ausgewählte Spiel soll eine Detailansicht eingeblendet werden, auf der die weiteren Eigenschaften verwaltet werden können.

Der Spieltag und der Beginn des Spiels sollen möglichst anwenderfreundlich über adäquate Steuerelemente realisiert werden.

Heim-, Auswärtsmannschaft und Saison sollen über Auswahlmöglichkeiten verknüpft werden können.

Automatische Generierung der Spiele (10 Punkte):

Der Anwender soll die Möglichkeit haben, alle Spiele für eine Saison (natürlich ohne die Ergebnisse) automatisch generieren zu lassen, sofern noch keine Spiele mit der Saison verknüpft sind.

Hierzu soll dem Anwender eine Schaltfläche zur Generierung aller Spiele einer Saison zur Verfügung stehen.

Stellen Sie für die Generierung einen eigenen Dialog bereit, in dem der Beginn der Saison (Datum) ausgewählt werden kann.

Generieren Sie für alle verknüpften Mannschaften jeweils die Spiele der Hin- und Rückrunde.

Legen Sie die Spiele auf die Wochenenden. Generieren Sie die Spielpaarungen zufällig nach folgendem Muster:

- 1 Spiel an den Freitagen um 20:30 Uhr
- 1 Spiel an den Sonntagen um 15:30 Uhr
- 1 Spiel an den Sonntagen um 17:30 Uhr
- alle restlichen Spiele an den Samstagen um 15:30 Uhr

Import von Spieltagen (5 Punkte):

Zusätzlich zur Eingabe über das Programm soll es möglich sein, Spieltage aus XML-Dateien zu importieren.

Das Datenformat der XML-Dateien kann der beiliegenden Datei „Spieltag 34.xml“ entnommen werden, die die Daten des 34. Spieltags der Saison 2016/2017 der 1. Fußballbundesliga enthält.

Bieten Sie in der Software die Möglichkeit zur Auswahl einer Import-Datei über einen Datei-Öffnen-Dialog (WPF) an.

Da die Import-Dateien unter Umständen nicht alle Pflichtangaben enthalten müssen, finden Sie sinnvolle Voreinstellungen für diese Fälle.

Rechnen Sie damit, dass Sie evtl. auch korrupte Import-Dateien bekommen könnten. Sichern Sie den Import-Prozess entsprechend ab.

Tippspiel (30 Punkte)

Nachfolgend werden die Anforderungen an die Tippspielfunktionalität beschrieben. Sowohl die Client- als auch die Serverseite ist gemäß den Anforderungen umzusetzen.

Anmeldung (5 Punkte):

Tipper (Spieler) sollen sich am Tippspiel-Client durch Eingabe ihres Spitznamens (Nicknames) anmelden können.

Der Spitzname wird ohne Beachtung von Groß- und Kleinschreibung überprüft (also wäre folgendes **äquivalent**: „Horst“, „horst“).

Ist die Anmeldung erfolgreich, wird der Tippspiel-Client geöffnet. Ist die Anmeldung nicht erfolgreich, wird der Anwender mittels einer klaren, benutzerfreundlichen Meldung über diesen Sachverhalt informiert.

Tabelle: Bettors

Tippsverwaltung (15 Punkte):

Realisieren Sie folgende Funktionalitäten zur Abbildung der Verwaltung von Tipps auf Spiele.

Alle (bereits vorhandenen) Spieltage einer Saison (diese soll auswählbar sein) sollen in einer Listenansicht chronologisch angezeigt werden.

Die Spieltage sollen auswählbar sein. Für den ausgewählten Spieltag soll eine Detailansicht eingeblendet werden, auf der für alle Spiele des Spieltags jeweils ein Tipp abgegeben werden kann. Die Informationen der einzelnen Spiele (z.B. welche Mannschaften gegeneinander spielen oder wann die Spiele stattfinden) sollen benutzerfreundlich angezeigt werden.

Schaffen Sie Möglichkeiten zur Anzeige, Neuanlage, zum Speichern, Bearbeiten und Löschen von Tipps.

Tipps zeichnen sich durch folgende Eigenschaften aus:

- Zeitpunkt (DateTime | Datum und Uhrzeit der Tippabgabe bzw. Änderung des Tipps)
- Heim-Tore (HomeTeamScore | Tore der Heim-Mannschaft)
- Auswärts-Tore (AwayTeamScore | Tore der Auswärts-Mannschaft)
- Spiel (MatchId | Verknüpfung zum Spiel-Datensatz)
- Tipper (BettorId | Verknüpfung zum angemeldeten Tipper)

Vergessen Sie nicht die Abbildung des Optimistic Lockings!

Tabelle: Bets

Spieltipps dürfen nur bis maximal 30 Minuten vor Spielbeginn abgegeben werden. Danach können keine neuen Spieltipps für das Spiel abgegeben werden. Bestehende Tipps können dann auch nicht mehr geändert oder gelöscht werden.

Der Zeitpunkt der Tippabgabe ist bei jedem Speichern automatisch zu ermitteln und zu persistieren.

Der Tipper wird automatisch über den angemeldeten Tipper gesetzt.

Aktuelle Tabelle der Saison (5 Punkte):

Über eine Listenansicht soll die aktuelle Tabelle einer Saison ausgegeben werden. Die Saison soll hierzu ausgewählt werden können.

Ebenfalls soll es möglich sein, den Spieltag auszuwählen, zu dem dann die Tabelle berechnet wird (Auswahlmöglichkeit „aktuell“ und Auswahl aller vorhandenen Spieltage).

Folgende Spalten sollen dargestellt werden:

- # (Platzierung)
- Team (Name der Mannschaft)
- Sp. (Anzahl Spiele)
- S (Anzahl Siege)
- U (Anzahl Unentschieden)
- N (Anzahl Niederlagen)
- +/- (Tordifferenz)
- Pkt. (Punkte)

Hinweise zur Berechnung:

- Für einen Sieg erhält eine Mannschaft 3 Punkte.
- Für ein Unentschieden erhält eine Mannschaft 1 Punkt.
- Für eine Niederlage erhält eine Mannschaft keine Punkte.

Tabellen: Seasons, Teams, SeasonsToTeamsRelation, Matches

Aktuelle Rangliste der Tipper (5 Punkte):

Über eine Listenansicht soll die aktuelle Rangliste der Tipper einer Saison ausgegeben werden.

Die Saison soll hierzu ausgewählt werden können.

Ebenfalls soll es möglich sein, den Spieltag auszuwählen, zu dem dann die Rangliste berechnet wird (Auswahlmöglichkeit „aktuell“ und Auswahl aller vorhandenen Spieltage).

Folgende Spalten sollen dargestellt werden:

- # (Platzierung)
- Spitzname
- Vorname
- Nachname
- Pkt. (Punkte)

Hinweise zur Berechnung:

- Der Tipper erhält 4 Punkte für das richtige Ergebnis des Spiels

- Der Tipper erhält 3 Punkte bei korrekter Tordifferenz des Spiels (Sieger muss richtig sein)
- Der Tipper erhält 2 Punkte, wenn er auf den richtigen Sieger des Spiels gesetzt hat.
- Andernfalls erhält der Tipper keine Punkte
- Der Tipper kann also maximal 4 Punkte für das exakt richtige Ergebnis erhalten. Stimmen nur die Tordifferenz und der Sieger, erhält er immer noch 3 Punkte. Stimmt nur der Sieger, erhält er noch 2 Punkte.

Tabellen: Seasons, Teams, SeasonsToTeamsRelation, Matches, Bettors, Bets

Dokumentation (10 Punkte):

Das Projekt und die Vorgehensweise sollen in einer kurzen Dokumentation vorgestellt werden.

Die Dokumentation soll das Oberflächen-Design vorstellen.

Bitte beschreiben Sie Ihre Beweggründe und was Sie mit dem Bedienaufbau des Softwaresystems erreichen wollen.

Die Dokumentation soll die Architektur des Softwaresystems beschreiben.

Fassen Sie die einzelnen Komponenten zu Bereichen zusammen und erläutern Sie diese.

Die Dokumentation soll aufführen, welche Aufgaben (und wie) bearbeitet wurden.

Die Dokumentation sollte (mit Bildern) ca. 10-15 DIN A4 Seiten umfassen.

Ausschließlich Bilder werden allerdings nicht akzeptiert.

Die Dokumentation muss als PDF oder als MS Word Dokument abgegeben werden.

Allgemeine Tipps zur Lösung der Projektaufgabe:

Lesen Sie zuerst alle Aufgaben durch und auch diese Liste mit Tipps. Sie sparen Zeit, wenn Sie zuerst alle Anforderungen sichten und bei der Implementierung der Aufgaben berücksichtigen. Dies ist besonders bei grundsätzlichen Entscheidungen wichtig!

Gehen Sie „datengetrieben“ vor.

Legen Sie für die notwendigen Daten das Datenmodell in Klassenform fest.

Stellen Sie danach die notwendigen Grundfunktionalitäten Richtung Datenbank sicher, bevor Sie die Anwendung als solche implementieren. (z.B. Laden, Speichern, Aktualisieren und Löschen der entsprechenden Daten)

Verschönerungen an der Oberfläche sollten Sie erst zum Schluss umsetzen.

Hängen Sie bei der Implementierung einer Teilfunktionalität, sollten Sie sich frühzeitig für die Implementierung der anderen Teilaufgaben entscheiden.

Haben Sie später noch genug Zeit, können Sie sich um die „Problemaufgabe“ kümmern.

Programmabstürze führen zu Punktabzug!

Rechnen Sie immer damit, dass Sie Fehleingaben erhalten. Prüfen Sie diese ab.

In der Realität kann es durchaus vorkommen, dass Sie fehlerhafte Daten für einen Datenimport erhalten.

Wenn Sie bei einigen geforderten Funktionalitäten nicht genau wissen, wie Sie diese implementieren können, steht es Ihnen frei, eine Suchmaschine Ihrer Wahl zu Rate zu ziehen. Hierbei hat sich die englische Sprache als besonders geeignet erwiesen.

Versuchen Sie die einzelnen Aufgaben vollständig zu lösen. Lösen Sie Aufgaben nicht nur zum Teil.

Vermeiden Sie Rechenübungen bzgl. ausreichender Punktzahl und der Anzahl bearbeiteter Aufgaben. Lösen Sie stattdessen die Aufgaben.