

Hausarbeit Wintersemester 2023/2024 (Block B)

Modul:	Objektorientierte Softwareentwicklung in C#
Modulgruppe:	MPO 2008: Basistechnologien (Informatik) (4,5 ECTS) MPO 2011: Basistechnologien (5 ECTS)
Prüfer:	Prof. Eicker
Ausgabe:	22.12.2023

Bearbeitungszeit:	22.12.2023 bis 28.04.2024 Eine Abgabe der Hausarbeit ist im genannten Zeitraum jederzeit möglich.
Anforderungen:	Die Hausarbeit besteht aus drei Aufgaben. Diese sind <u>alle</u> gemäß den Anforderungen auf den nachfolgenden Seiten zu bearbeiten.
Abgabe:	Die Abgabe erfolgt in Moodle (LMS).

Viel Erfolg!

Aufgabenstellung Hausarbeit im Kurs Objektorientierte Softwareentwicklung in C#

Kurzübersicht und Einführung in die Hausarbeit

Einordnung

Im Rahmen der Hausarbeit ist ein **C#-Projekt** auf Basis einer vorgegebenen Anforderungsbeschreibung zu erstellen. Die Anforderungsbeschreibung umfasst funktionale Anforderungen sowie Vorgaben bzgl. dessen Umsetzung mittels spezifischer Konzepte.

Neben der praktischen Umsetzung des C#-Projekts sind zusätzlich **zwei theoretische Fragen** schriftlich zu bearbeiten.

Die genaue Aufgabenstellung finden Sie weiter hinten im Dokument.

Form der Hausarbeit

Auf Grund der zweigeteilten Aufgabenstellung sind daher **zwei Ergebnisformate** einzureichen:

1. Das Ergebnis der praktischen Aufgabe mit folgenden Bestandteilen:
 - a. Die **C#-Projektmappe** (also der gesamte Projektordner),
 - b. eine kurze **Dokumentation** als PDF-Dokument, in der die verwendete Entwicklungsumgebung, Besonderheiten und Probleme während der Entwicklung sowie getroffene Annahmen festgehalten werden,
 - c. das generierte **Build-Ergebnis** als .exe-Datei (für den Fall, dass die Kompilierung bei der Bewertung nicht funktionieren sollte),
 - d. **Beispieldaten** im XML-Format mit denen die Funktionalität überprüft werden kann.
2. Die theoretischen Fragen sind textuell zu beantworten. Die Abgabe erfolgt als **PDF-Dokument** auf Basis der bereitgestellten Vorlage.

Anforderungen an die C#-Projektmappe

Ziel des C# Projekts ist es zu prüfen, ob die **Lernziele der Veranstaltung** erreicht wurden. Die Studierenden sollen daher zeigen, dass Sie die geforderten Konzepte sinnvoll und zielführend einzusetzen können. Eine perfekt funktionierende Software, die Anforderungen über die Aufgabenbeschreibung hinaus erfüllt (z.B. Berücksichtigung von Usability-Aspekten, umfassendes Error-Handling etc.), wird nicht erwartet.

Der Quellcode sollte sich an den im Kurs enthaltenen **Code-Styles** orientieren.

Als Zielframework ist **.NET 6.0** (LTS) zu nutzen. Auch sollte der Code durch **Kommentare** unterstützt werden, um einzelne Methoden oder besondere Vorgehensweisen zu dokumentieren.

Zudem sollten möglichst die jeweils **passenden Konzepte** eingesetzt werden. Bewertet werden dabei auch die **Wartbarkeit, Zuverlässigkeit und Korrektheit** (z.B. wie groß der Aufwand für Änderungen oder Anpassungen ist) des Codes.

Externe Bibliotheken können optional eingebunden werden, sofern diese öffentlich zugänglich sind (GitHub etc.), dessen Zweck in der Dokumentation vermerkt ist und die Stellen, an denen diese zum Einsatz kommen klar gekennzeichnet sind. Die Aufgabenstellung ist ohne die Einbindung externer Bibliotheken vollständig lösbar.

Anforderungen an die theoretischen Fragen

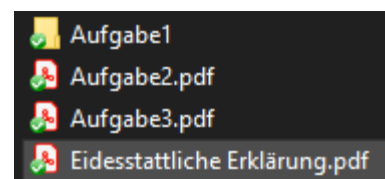
Die Fragen sind in einem **Umfang von jeweils ca. 1000-1500 Wörtern** (pro Aufgabe) zu beantworten. Der Fokus sollte dabei auf einem zielführenden Vorgehen liegen. Es sollten geeignete Quellen (z.B. Online-Quellen, Lehrbücher oder wissenschaftliche Paper) zitiert werden, um die eigenen Aussagen zu belegen. Auf Grund der aktuellen Entwicklungen im Bereich KI-generierter Texte wird der Quellenarbeit bei der Begutachtung der Leistung eine wesentliche Bedeutung beigemessen. Sprache und Argumentation sollen wissenschaftlichen Ansprüchen genügen.

Abgabe

Den Abgabetermin für die Hausarbeit finden Sie auf der ersten Seite dieses Dokuments sowie im Moodle-Kurs. Eine frühere Abgabe ist selbstverständlich möglich. Die fertige Hausarbeit ist unter „Abgabe Hausarbeit“ im Kurs hochzuladen.

Für die Abgabe und Organisation der Dateien gelten die folgenden Anforderungen:

- Ein Ordner „**Aufgabe 1**“ enthält
 - die **C#-Projektmappe**,
 - die **Dokumentation** als PDF-Dokument,
 - innerhalb der C#-Projektmappe enthält der Ordner *bin/debug* den finalen Stand der ausführbaren **Build-Dateien**,
 - innerhalb der C#-Projektmappe befindet sich mindestens eine .xml-Datei mit **Beispieldaten**, die beim Start des Programms automatisch geladen werden.
- Eine PDF-Datei „**Aufgabe 2.pdf**“ enthält die Lösung der ersten theoretischen Aufgabe.
- Eine PDF-Datei „**Aufgabe 3.pdf**“ enthält die Lösung der zweiten theoretischen Aufgabe.
- Die ausgefüllte und unterschriebene Eidesstattliche Erklärung als PDF-Datei „**Eidesstattliche Erklärung.pdf**“
- Für den Upload alle Dateien und Ordner in ein **Zip-Archiv** legen, so dass nur eine Datei hochgeladen werden muss
 - Dateiname: "**<Kurskürzel>_<Semester>_Hausarbeit_**" + **Nachname** + **".zip"** verwenden - wobei 'Nachname' durch den Nachnamen des Autors zu ersetzen ist, also z.B. OBJC_WS2023_Hausarbeit_Brogt.zip



Aufgabe 1: C#-Projekt (60 Punkte)

Aufgabe

Das Ziel des C#-Projekts ist der Entwurf sowie die Implementierung einer Software für einen **Werkzeugverleih**. Die Software soll die Abwicklung der Ausleihe verschiedener Werkzeuge (Bohrmaschine, Rasenmäher etc.) technisch unterstützen, indem sie wesentliche Informationen verwaltet.

Die nachfolgenden Anforderungen wurden gemeinsam mit dem beauftragenden Unternehmen, der „Tool2Go GmbH“, erarbeitet und bereits implementierungsnah beschrieben. Die Beschreibung ist dabei nicht vollständig und es ist zum Teil notwendig, weitere Klassen, Properties und Methoden einzuführen. Wesentlich ist, dass die unten angeführten Anforderungen korrekt berücksichtigt und umgesetzt werden.

Die Bedienung der Software soll über die Konsole erfolgen.

Zudem soll eine Speicherung mittels XML-Serialisierung ermöglicht werden. Eine anderweitige Speicherung in einer lokalen Datenbank oder Ähnlichem ist nicht erforderlich. Es ist zudem ausreichend, wenn der Benutzer den Speicher-Befehl explizit auswählen muss, um den gesamten aktuellen Datenbestand zu speichern. Die Speicherung kann in einer zentralen oder mehreren einzelnen XML-Dateien erfolgen.

Zur Realisierung der Datenstruktur sollen sinnvolle Datentypen gewählt werden (*int*, *String*, *DateTime* etc.)

Die Aufgabenstellung ist bewusst offengehalten, um eigene Ansätze und Entscheidungen zu ermöglichen. Falls nötig können **Annahmen** getroffen werden. Diese können dann in der Dokumentation festgehalten werden.

Komponenten und Anforderungen

Kundenverwaltung (12 Punkte)

- Funktionen:
 - Kunde anlegen, bearbeiten und löschen
 - Textuelle Ausgabe der Kundendaten
- Datenstruktur
 - Kunde: Vorname, Nachname, Adresse, Geburtsdatum, Bankverbindung, Privat- oder Geschäftskunde

Werkzeugverwaltung (12 Punkte)

- Funktionen:
 - Werkzeugkategorie (Bohrmaschine, Rasenmäher etc.) anlegen, bearbeiten und löschen
 - Werkzeug in Werkzeugkategorie anlegen, bearbeiten und löschen

- Textuelle Ausgabe der Werkzeugkategorien und Werkzeuge
- Datenstruktur
 - Werkzeugkategorie: Name, Beschreibung, Preis pro Tag, Preis pro Woche
 - Werkzeug: Hersteller (z.B. Bosch), Bezeichnung (z.B. GBM 16-2 RE), Stromversorgung (Akku, Kabel oder stromlos)

Verleihverwaltung (16 Punkte)

- Funktionen:
 - Ausleihe anlegen, bearbeiten und löschen
 - Ausleihe einem Kunden und einem Werkzeug aus einer Werkzeugkategorie zuordnen
 - Textuelle Ausgabe der Ausleihen
- Datenstruktur
 - Ausleihe: Kunde, Liste von geliehenen Werkzeugen (ein Kunde kann pro Leihvorgang mehrere Werkzeuge leihen), Verleihbeginn, Verleihdauer, Gesamtkosten

Weitere Anforderungen (20 Punkte)

- Weitere **benötigte Datenstrukturen sollen nach Bedarf** angelegt und verwendet werden. Auch sollen die angegebenen Datenstrukturen (Kunde, ...) um weitere sinnvolle Daten ergänzt werden.
- Die Software ist so zu implementieren, dass die **Bedienung über die Konsole** (z.B. nummerierte Menüpunkte und Auswahl über Eingabe) erfolgt. Eine andere Form der GUI (z.B. Windows Forms) wird nicht gewertet.
- Die eingegebenen **Daten** sollen auch nach Beendigung der Software zur Verfügung stehen und sind daher zu **speichern**. Dies soll mit Hilfe der XML-Serialisierung erfolgen.
- Folgende nicht-funktionale Anforderungen sind besonders bei der Entwicklung zu beachten und sollten durch den Einsatz entsprechender Konzepte (Abstraktion, Generics, ...) berücksichtigt werden: **Wartbarkeit, Zuverlässigkeit und Korrektheit**.

- Ende Aufgabe 1 -

Aufgabe 2: Neuerungen in C# 11.0 und C# 10.0 (15 Punkte)

Aufgabe

Stellen Sie vier Neuerungen vor, die mit C# 11.0 oder C# 10.0 eingeführt wurden. Gehen Sie dabei jeweils auf die Veränderungen und die damit einhergehenden Vor- und Nachteile für den Einsatz in C# ein. Ergänzen Sie Ihre Ausführungen gerne durch entsprechende Code-Beispiele.

Anforderungen

Die textuellen Ausführungen sollen in sich geschlossen sein und dem **wissenschaftlichen Anspruch** genügen. Nutzen Sie zur Beantwortung der Frage daher geeignete Quellen (die natürlich angegeben werden sollten) und gliedern Sie Ihre Ausführungen in einer geeigneten Art und Weise.

Ein Inhalts- Abbildungs- und Tabellenverzeichnis ist nicht notwendig. Ein Literaturverzeichnis sollte angefügt werden. Sie finden eine entsprechende Word-Vorlage im Moodle-Kurs.

Beachten Sie den zuvor angeführten Umgang von ca. 1000-1500 Wörtern zur Beantwortung der Frage (ohne Deckblatt und Literaturverzeichnis).

- Ende Aufgabe 2 -

Aufgabe 3: Grundlagen .NET (15 Punkte)

Aufgabe

Stellen Sie .NET ausführlich vor und gehen Sie auf die verschiedenen Bestandteile des Frameworks und deren Aufgaben ein (z.B. MISL). Sie können Ihre Ausführungen durch geeignete Abbildungen oder Tabellen ergänzen.

Geben Sie zudem einen Ausblick darauf, wie die geplante Weiterentwicklung von .NET aussehen soll.

Anforderungen

Die textuellen Ausführungen sollen in sich geschlossen sein und dem **wissenschaftlichen Anspruch** genügen. Nutzen Sie zur Beantwortung der Frage daher geeignete Quellen (die natürlich angegeben werden sollten) und gliedern Sie Ihre Ausführungen in einer geeigneten Art und Weise.

Ein Inhalts- Abbildungs- und Tabellenverzeichnis ist nicht notwendig. Ein Literaturverzeichnis sollte angefügt werden. Sie finden eine entsprechende Word-Vorlage im Moodle-Kurs.

Beachten Sie den zuvor angeführten Umgang von ca. 1000-1500 Wörtern zur Beantwortung der Frage (ohne Deckblatt und Literaturverzeichnis).

- Ende Aufgabe 3 -

- Ende Aufgabenstellung Hausarbeit -