

Praktische Informatik

Vorlesung 00

Organisatorisches

Inhalt

- Vorstellung des Dozenten
- Lernziele und Themen
- Aufbau der Veranstaltung
- Literatur
- Prüfung und Bewertung
- Erwartungen

Ihr Dozent

- Prof. Dr.-Ing Alexander Stuckenholtz
 - Lehrgebiet praktische Informatik
 - Büro: H4.3-E02-210
- Sprechstunde nach Vereinbarung per E-Mail
 - alexander.stuckenholtz@hshl.de
- Infos
 - <http://hshl.de/vita-prof-dr-ing-alexander-stuckenholtz/>
 - https://www.xing.com/profile/Alexander_Stuckenholtz

Lernziele

- **Ein umfangreiches objektorientiertes Framework für die Anwendungserstellung nutzen können.**
- Grafische Benutzeroberflächen in C# mit WPF entwickeln können.
- Das Konzept von XAML und Code Behind verstehen.
- Controls zur Nutzerinteraktion benutzen können.
- Layouts benutzen können.
- Data Binding verstehen und benutzen können.
- Graphikorientierte Programme mit WPF realisieren können.
- Nebenläufige Programme realisieren können.
- Netzwerkkommunikation umsetzen können.
- Grundlagen der C++ Programmierung beherrschen.

Aufbau der Veranstaltung

- **Diese Veranstaltung findet wieder in Präsenz statt!**
- Vorlesung
 - Die Termine sind auf den nächsten Folien dargestellt.
 - Bitte beachten Sie, dass die Vorlesung nach Weihnachten erst ab dem 10.1.2022 wieder weiter geht.
 - Zusätzlich können Sie weiterhin die Lehrvideos und Folien in der Lernplattform finden.
- Übung
 - Die Übung findet Mittwochs ab 14:00 Uhr in H4.2-E00-130/140 statt.
 - Es gibt zwar Übungsaufgaben, aber meistens wurden die Übungen dazu genutzt, am eigenen Semesterprojekt zu arbeiten.

Themen

#	Datum	Uhrzeit	Ort	Thema
1	23.09.2022	9:30 Uhr	H3.3-E00-010	Orga + Frameworks und GUIs
2	...	9:30 Uhr	H3.3-E00-010	Projektmanagement
3	...	9:30 Uhr	H3.3-E00-010	Delegates und Ereignisse
4	...	9:30 Uhr	H3.3-E00-010	Einführung in die WPF
5		9:30 Uhr	H3.3-E00-010	Einfache Steuerelemente
6		9:30 Uhr	H3.3-E00-010	Fenster und Layouts
7		9:30 Uhr	H3.3-E00-010	2D Grafik
8		9:30 Uhr	H3.3-E00-010	Datenbindung
9		9:30 Uhr	H3.3-E00-010	Styling
10		9:30 Uhr	H3.3-E00-010	Listen, Tabellen und Bäume
11		9:30 Uhr	H3.3-E00-010	Xamarin
12		9:30 Uhr	H3.3-E00-010	Nebenläufigkeit
13		9:30 Uhr	H3.3-E00-010	Netzwerkprogrammierung
14		9:30 Uhr	H3.3-E00-010	Programmierung in C++
15		9:30 Uhr	H3.3-E00-010	C++ Standardbibliothek

Literatur

- Andreas Kühnel: C# 6 mit Visual Studio, 7. Auflage, Rheinwerk Verlag, Bonn, 2016.
- Thomas Claudius Huber: Windows Presentation Foundation – Das umfassende Handbuch, 4. Auflage, Rheinwerk Verlag, Bonn, 2016.

Prüfung

- Wie ist die Prüfungsform?
 - Dieses mal keine Klausur!
- Sondern: Eigenständige Programmierung einer Anwendung.
 - Ziel des Projekts: Beweis, dass die Lerninhalte verstanden wurden.
 - Es wird eine Liste von möglichen Aufgaben in die Lernplattform eingestellt.
 - Eigene Ideen sind nach Absprache ebenfalls möglich und gerne gewünscht!
- Das Projekt wird in der Prüfungsphase in einer mündlichen Prüfung vorgestellt.
 - Der Programmcode muss dabei **selbst erklärt werden können**.
 - Das Projekt wird am letzten Vorlesungstag vollständig eingereicht.
- Wichtig: Das Programm wird ohne Hilfe vollständig eigenständig erstellt.
 - Die Verwendung von fremdem Programmcode oder Helfern ist **Betrug!!!**
 - Kein Quellcode aus Github, Youtube-Videos oder aus sonst irgendeiner Quelle!!!
 - Es dürfen ausschließlich Bestandteile der .Net-Bibliothek verwendet werden.
- **Wichtig:**
 - **Jeder Studierender meldet mir (ohne weitere Nachfrage) spätestens 4 Wochen nach Vorlesungsbeginn, welches Projekt er/sie erstellen will.**
 - **Das Projekt wird mir in der Vorlesungsphase einmal kurz gezeigt! Das ist ein Sicherheitsmechanismus für Sie! In der Prüfung kann ich nur noch eine Note vergeben, keine Beratung mehr machen!**

Bewertungskriterien

1. Die folgenden Bewertungskriterien haben eine **Rangfolge**:
2. Korrekter C# Code.
 - Übersetzbar und es tut was es soll.
3. Komplexität des Projektes
 - TicTacToe ist nun mal einfacher, als ein OpenStreetMap-Parser.
4. Nutzung objektorientierter Prinzipien
 - Ist alles in eine Klasse „reingeschmiert“, wird entsprechend schlecht bewertet!
 - Vernünftiges Datenmodell → Man sollte das auch mal in UML darstellen können.
 - Sinnvolle Aufteilung auf Klassen und Methoden.
 - Wenn die meisten Methoden static sind, wurde OOP nicht benutzt!
5. Zeigen, dass man Lerninhalte der Vorlesung beherrscht
 - WPF, XAML, Datenbindung, 2D-Grafik, Nebenläufigkeit, Netzwerkprogrammierung, ...
 - Desto weniger man verwendet, desto schlechter wird bewertet!
 - Wenn man das Projekt in C++/Windows Forms/ oder sonst irgendeiner Sprache bearbeitet, die wir nicht in der Vorlesung behandeln, kann es nur eine 5,0 geben!
6. Einhalten von Qualitätskriterien
 - Gute Usability und look-and-feel!
 - Sinnvolle Benennung und durchgängig gute Formatierung des Quellcodes.
 - Keine elends-langen Methoden oder Klassen.
 - Die Königsdisziplin: Unit-Tests!

Was ich erwarte...

- Nehmen Sie die Veranstaltung ernst.
 - Das hier ist mehr, als nur lästige Pflicht.
 - Sie sollen etwas lernen, was Ihnen später im Beruf helfen wird.
- Entsprechend...
 - Beschäftigen Sie sich mit dem Inhalt!
 - Spielen Sie mit den Dingen, die ich Ihnen zeige!
 - Fangen Sie **rechtzeitig** an, das Projekt umzusetzen!

Was Sie erwarten können...

- Die Folien stellen ein durchdachtes Script dar!
 - Ich werde mir mit den Lehrvideos echt Mühe geben!
- Unterbrechen Sie mich **jederzeit**, wenn Sie Fragen haben.
- Nehmen Sie Kontakt auf!
 - Kommen Sie in meine Sprechstunde!
 - Schreiben Sie mir Mails!