|  |
| --- |
| **Plattformunabhängige Programmierung in C#** |
| Projekt Mono als Konkurrenz für das Java Runtime Environment |
|  |
| Diese Ausarbeitung befasst sich mit der Plattformunabhängigen Programmierung in C#, mittels Projekt Mono, einer Open Source Implementierung des proprietären .NET Frameworks und deren Konkurrenzstellung zum Java Runtime Environment. |
|  |
| **Jens Schwanke** |
| **19.04.2008** |
|  |

Inhaltsverzeichnis

[1 Einleitung 5](#_Toc199693496)

[2 Material und Methoden 6](#_Toc199693497)

[2.1 Grundlagen 6](#_Toc199693498)

[2.1.1 Das .NET Framework 6](#_Toc199693499)

[2.1.2 Projekt Mono 6](#_Toc199693500)

[2.1.3 C# 6](#_Toc199693501)

[2.1.4 Java 6](#_Toc199693502)

[2.1.5 C++ 6](#_Toc199693503)

[2.1.6 Managed C++ 6](#_Toc199693504)

[2.1.7 Dynamische Bibliotheken 6](#_Toc199693505)

[2.2 Versuchsapplikation 6](#_Toc199693506)

[2.2.1 Versuchsaufbau 6](#_Toc199693507)

[2.2.2 Aufbau der Applikation 6](#_Toc199693508)

[2.2.3 Versuchsdurchführung 6](#_Toc199693509)

[2.2.4 Quantitative Betrachtung 6](#_Toc199693510)

[2.2.5 Qualitative Betrachtung 6](#_Toc199693511)

[2.3 Erweiterung der Versuchsapplikation 6](#_Toc199693512)

[2.3.1 Versuchsaufbau 6](#_Toc199693513)

[2.3.2 Aufbau der Applikation 6](#_Toc199693514)

[2.3.3 Versuchsdurchführung 6](#_Toc199693515)

[2.3.4 Quantitative Betrachtung 6](#_Toc199693516)

[2.3.5 Qualitative Betrachtung 6](#_Toc199693517)

[3 Ergebnisse 7](#_Toc199693518)

[3.1 Ergebnisse der Versuchsapplikation 7](#_Toc199693519)

[3.1.1 Quantitative Ergebnisse 7](#_Toc199693520)

[3.1.2 Qualitative Ergebnisse 7](#_Toc199693521)

[3.2 Ergebnisse der erweiterten Versuchsapplikation 7](#_Toc199693522)

[3.2.1 Quantitative Ergebnisse 7](#_Toc199693523)

[3.2.2 Qualitative Ergebnisse 7](#_Toc199693524)

[3.3 Zusammenfassung 7](#_Toc199693525)

[3.3.1 Quantitative Betrachtung der Ergebnisse 7](#_Toc199693526)

[3.3.2 Qualitative Betrachtung der Ergebnisse 7](#_Toc199693527)

[4 Diskussion 8](#_Toc199693528)

[5 Zusammenfassung 9](#_Toc199693529)

[6 Abbildungsverzeichnis 10](#_Toc199693530)

[7 Literaturverzeichnis 11](#_Toc199693531)

[8 Referenzen 12](#_Toc199693532)

[9 Eidesstattliche Erklärung 13](#_Toc199693533)

# Einleitung

Diese Ausarbeitung, welche im Rahmen meiner Studienarbeit entstanden ist, befasst sich mit der plattformübergreifenden beziehungsweise plattformunabhängigen Programmierung in C#. Die Programmiersprache C# ist eine von Microsoft entwickelte proprietäre objektorientierte Sprache, welche im Rahmen des .NET Frameworks im Jahre 2000 veröffentlicht wurde. Im gleichen Jahr lies Microsoft ein Teil des .NET Frameworks, um generauer zu sein, die „Common Language Infrastructure (CLI)“, als auch die Sprachspezifikation von C# bei der „International Organization for Standardization“ und bei der „European Computer Manufacturers Association“ standardisieren, dies ebnete den Weg für eine freie und plattformübergreifende Implementierung des CLI Standards. Weitere Grundlegende Informationen zu dem .NET Framework oder C# werden Sie im Grundlagen Kapitel finden.

Die wohl am Meisten bekannteste und verbreiteteste offene Implementierung des CLI Standards ist das Mono Projekt. Es wird unter Führung von Novell entwickelt, welche wiederrum ein Patent Abkommen mit Microsoft abgeschlossen haben, welches einen Patentschutz für das Mono Projekt beinhaltet.

Anhand des Mono Projekts, werde ich zeigen, dass eine plattformunabhängige Programmierung in C#, mittels der CLI Implementierung von Mono möglich ist. Des Weiteren will ich die Möglichkeiten und die Grenzen einer plattformunabhängigen Implementierung mittels des Mono Projekts offenbaren. Dies wird soll beispielhaft an einem Test Programm gezeigt werden, hierzu wird im ersten Versuch ein CT-Picture Viewer in C# implementiert. Es existieren bereits Implementierungen eines solchen Viewers in Java und C++, welche durch den Studiengang Medizinische Informatik an der Universität Heidelberg und Hochschule Heilbronn entstanden sind. An Hand des oben erwähnten CT-Picture Viewers, werde ich versuchen die Unterschiede und die Gemeinsamkeiten von Java und C# auf der Ebene der plattformunabhängigen Programmierung herauszuarbeiten.

In einem zweiten Versuch, werde ich Teile des CT-Viewers nativ in C++ implementieren und auf diese native Implementierung mittels Managed C++ zugereifen. Dieser Versuch soll die Möglichkeiten von C# bzw. dem .NET Framework ausreizen und er soll zeigen welchen Performance Unterschied eine teilweise native Implementierung gegenüber einer reinen C# Implementierung hat.

Ein detaillierter Versuchsaufbau der beiden Versuchsapplikationen wird im folgenden Kapitel näher beschrieben. Die Ergebnisse und die Auswertung der Ergebnisse werden in den darauf folgenden Kapiteln behandelt.

Das letzte Kapitel fasst die Kerngedanken dieser Ausarbeitung zusammen und setzt sie in Relation zu den Ergebnissen, welche durch die Versuche hervorgegangen sind.

# Material und Methoden

## Grundlagen

### Das .NET Framework

### Projekt Mono

### C#

### Java

### C++

### Managed C++

### Dynamische Bibliotheken

## Versuchsapplikation

### Versuchsaufbau

### Aufbau der Applikation

### Versuchsdurchführung

### Quantitative Betrachtung

### Qualitative Betrachtung

## Erweiterung der Versuchsapplikation

### Versuchsaufbau

### Aufbau der Applikation

### Versuchsdurchführung

### Quantitative Betrachtung

### Qualitative Betrachtung

# Ergebnisse

## Ergebnisse der Versuchsapplikation

### Quantitative Ergebnisse

### Qualitative Ergebnisse

## Ergebnisse der erweiterten Versuchsapplikation

### Quantitative Ergebnisse

### Qualitative Ergebnisse

## Zusammenfassung

### Quantitative Betrachtung der Ergebnisse

### Qualitative Betrachtung der Ergebnisse

# Diskussion

# Zusammenfassung

# Abbildungsverzeichnis

# Literaturverzeichnis

Kühnel, A. (2006). *Visual C# 2005.* Galileo Computing.

# Referenzen

# Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die Studienarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, alle Ausführungen, die anderen Schriften wörtlich oder sinngemäß entnommen wurden, kenntlich gemacht sind und die Arbeit in gleicher oder ähnlicher Fassung noch nicht Bestandteil einer Studien- oder Diplomarbeit war.

Unterschrift des Verfassers