

# Technology Arts Sciences TH Köln

Programmierung graphischer Benutzeroberflächen  
Bericht zum Semesterprojekt SS2021

Mahdi Bagheri  
Matrikelnummer: [REDACTED]

1. September 2021

## **Inhaltsverzeichnis**

|          |                                   |           |
|----------|-----------------------------------|-----------|
| <b>1</b> | <b>Einleitung</b>                 | <b>3</b>  |
| <b>2</b> | <b>Arbeitsumgebung</b>            | <b>3</b>  |
| <b>3</b> | <b>Java Applikation mit Swing</b> | <b>4</b>  |
| <b>4</b> | <b>C/C++ mit WINAPI</b>           | <b>8</b>  |
| <b>5</b> | <b>AWT oder Swing?</b>            | <b>12</b> |
| <b>6</b> | <b>Java oder C/C++ ?</b>          | <b>12</b> |
| <b>7</b> | <b>Erklärung</b>                  | <b>14</b> |

# 1 Einleitung

Der nachfolgende Bericht umfasst die Bearbeitung der Projektaufgaben, in dem Modul GUI - Programmierung graphischer Benutzeroberflächen.

Dieser besteht aus den Ergebnissen und Vorstellungen der beiden zu erarbeitenden graphischen Benutzeroberflächen, von denen eine mit der Programmiersprache Java (mit AWT oder SWING) und die andere mit den Sprachen C/C++ (mit der Windows WINAPI) umgesetzt wurde.

Im Anschluss daran werden einmal die Argumente für die Nutzung von C/C++ oder Java gegenüber der anderen Sprache behandelt, ebenso die Argumente, sich entweder für AWT oder Swing zu entscheiden im Fall von Java.

# 2 Arbeitsumgebung

Für die Bearbeitung der Aufgaben wurde ein Rechner genutzt der mit Windows 8.1 betrieben wird.

Als Programmierumgebung wird wie in den Aufgaben vorgegeben Code Blocks in der Version 20.03 mit dem gcc8.10 Compiler 32 Bit genutzt, um die C/C++ Aufgabe umzusetzen.

Für die Umsetzung der Java Aufgabe wurde die IntelliJ IDEA Community Edition verwendet in der Version 2021.1 mit OpenJDK 14.

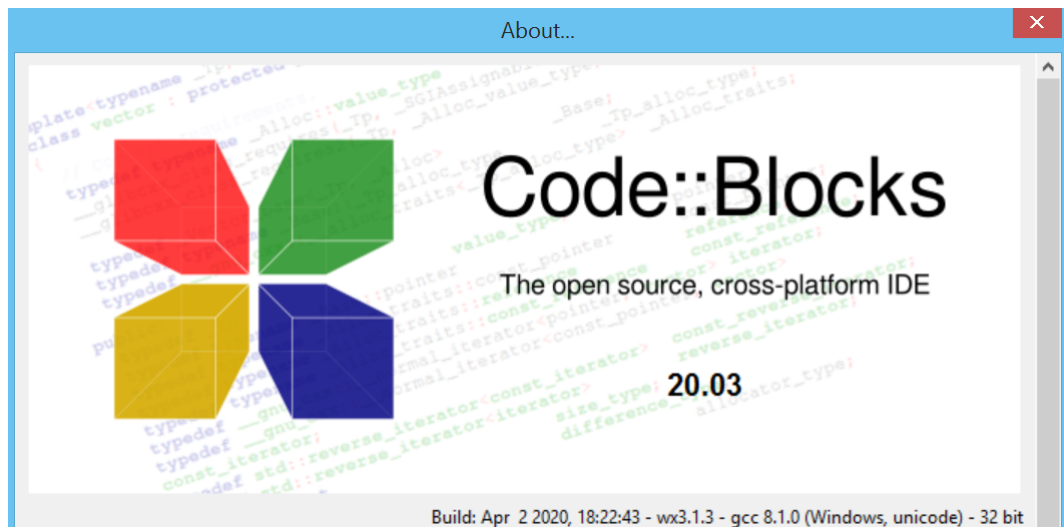


Abbildung 1: Code::Blocks 20.03 gcc8.1.0 32 bit



Abbildung 2: IntelliJ IDEA 2021.1 (Community Edition)

### 3 Java Applikation mit Swing

Für die Java Applikation sollte eine graphische Anwendung erstellt werden, mit der es möglich ist zu einer eingegebenen Zahl eine Prüfziffer zu ermitteln oder eine Zahl auf ihre Gültigkeit zu prüfen. Die Luhn Formel in dieser Anwendung arbeitet nach dem Modulo 10 Konzept.

Beim Start der Anwendung öffnet sich ein leeres Fenster mit zwei Menüpunkten "Datei" und "Luhn-Formel-Test". Der Menüpunkt "Datei" hat als einziges einen Unterpunkt "Beenden", über den das Programm beendet werden kann.

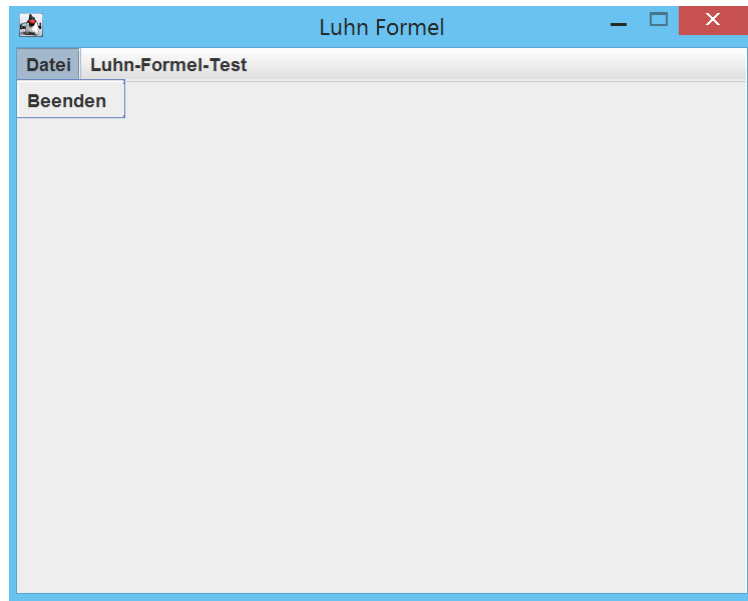


Abbildung 3: Java Anwendung nach Start

Wählt man den Menüpunkt "Luhn-Formel-Test" aus, öffnet sich ein weiteres Fenster.

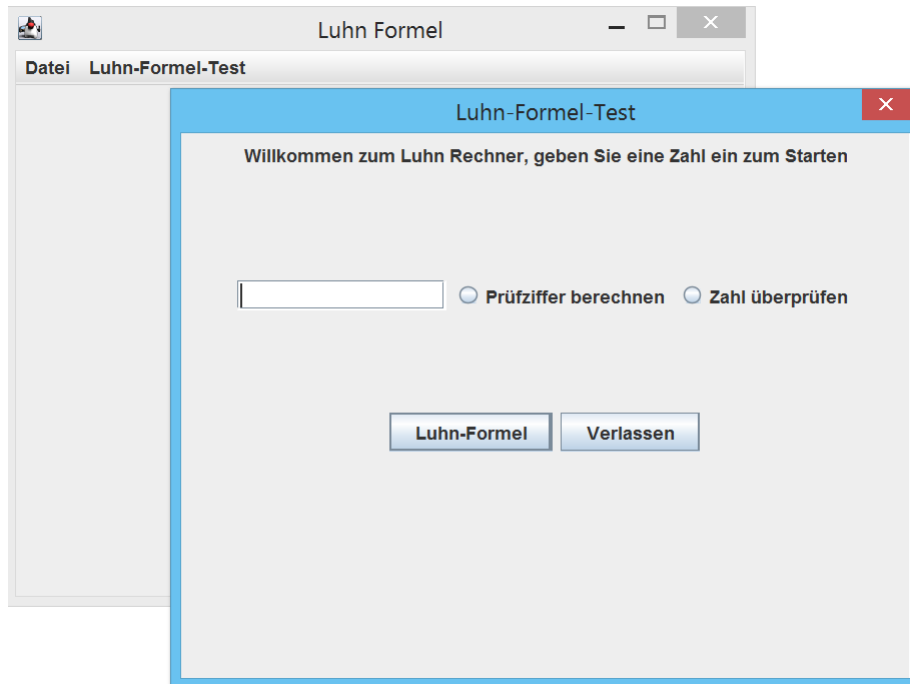


Abbildung 4: Luhn-Formel-Test Fenster

In diesem Fenster gibt es ein Textfeld in das man eine Zahl eingeben kann. Mit den nebenstehenden Radio-Buttons kann weiterhin ausgewählt, ob zu der eingegebenen Zahl die Prüfziffer berechnet werden soll, oder die Zahl auf ihre Gültigkeit überprüft werden soll. Um die Berechnung zu beginnen, betätigt man im Anschluss den Button mit der Bezeichnung "Luhn-Formel". Die Betätigung der Enter Taste ermöglicht ebenfalls eine Ausführung, sofern der Nutzer sich im Eingabefeld befindet. Ist das Textfeld leer, keine Zahl eingegeben worden oder kein Radio Button ausgewählt, erhält der Nutzer ein kurzes Feedback im Fenster, was daraufhin hinweist.

The dialog box is titled "Luhn-Formel-Test" and contains the following elements:

- Header: "Willkommen zum Luhn Rechner, geben Sie eine Zahl ein zum Starten"
- Input field: Contains the number "1234".
- Radio buttons: "Prüfziffer berechnen" (selected) and "Zahl überprüfen".
- Buttons: "Luhn-Formel" and "Verlassen".
- Output: "Prüfziffer: 4"

Abbildung 5: Prüfzifferberechnung zu einer eingegebenen Zahl

The dialog box is titled "Luhn-Formel-Test" and contains the following elements:

- Header: "Willkommen zum Luhn Rechner, geben Sie eine Zahl ein zum Starten"
- Input field: Contains the number "12344".
- Radio buttons: "Prüfziffer berechnen" and "Zahl überprüfen" (selected).
- Buttons: "Luhn-Formel" and "Verlassen".
- Output: "Luhn Prüfsumme: 0. Der Wert ist gültig"

Abbildung 6: Überprüfung der Gültigkeit zu einer eingegebenen Zahl

Der Verlassen-Button sorgt dafür, dass das aktuelle Dialog-Fenster geschlossen wird.

## 4 C/C++ mit WINAPI

Die folgende Anwendung wurde mittels C/C++ erstellt unter Nutzung der WINAPI. Die Anwendung ermöglicht dem Nutzer eine .txt oder .asc Datei auszuwählen und dessen Inhalt im Fenster anzeigen zu lassen. Zeitgleich ändert sich der Titel der Fensters zum ausgewählten Dateinamen.

Startet man die Anwendung ist ein leeres Fenster zu sehen mit zwei Menüpunkten "Datei" und "Text einlesen". Auch hier besitzt der Menüpunkt "Datei" als einziges einen Unterpunkt "Beenden", welcher bei Ausführung zum Beenden der Anwendung führt. Weiterhin ist ganz unten im Fenster ein Button, der ebenfalls bei Auswahl zum Beenden der Anwendung führt.

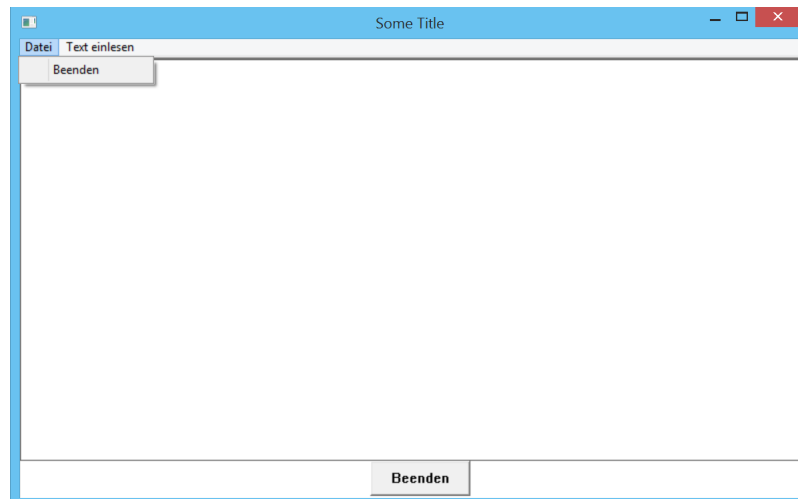


Abbildung 7: C/C++ Anwendung nach Programmstart

Wird der Menüpunkt "Text einlesen" ausgewählt, öffnet sich ein Dialogfenster, welche dem Nutzer ermöglicht eine .txt oder .asc Datei auszuwählen.



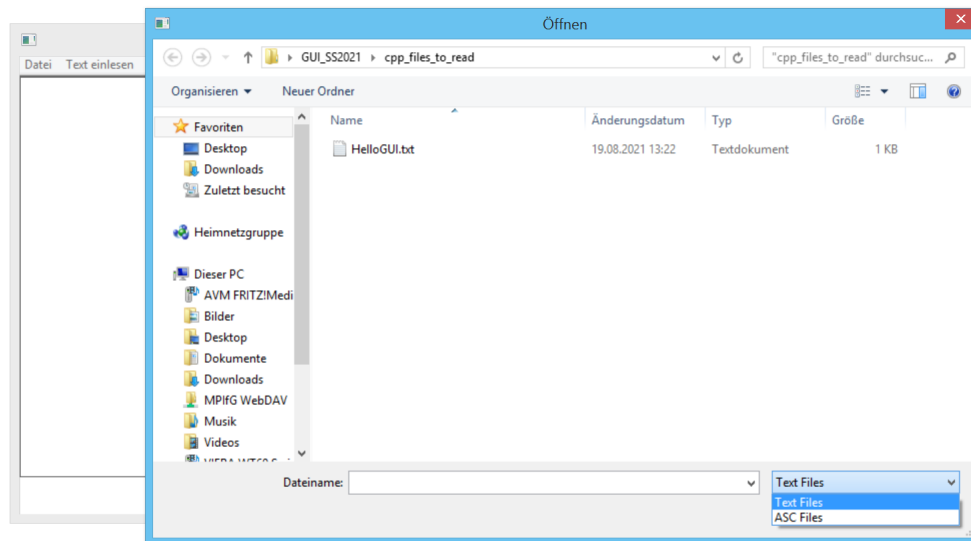


Abbildung 8: Auswählen einer .txt Datei

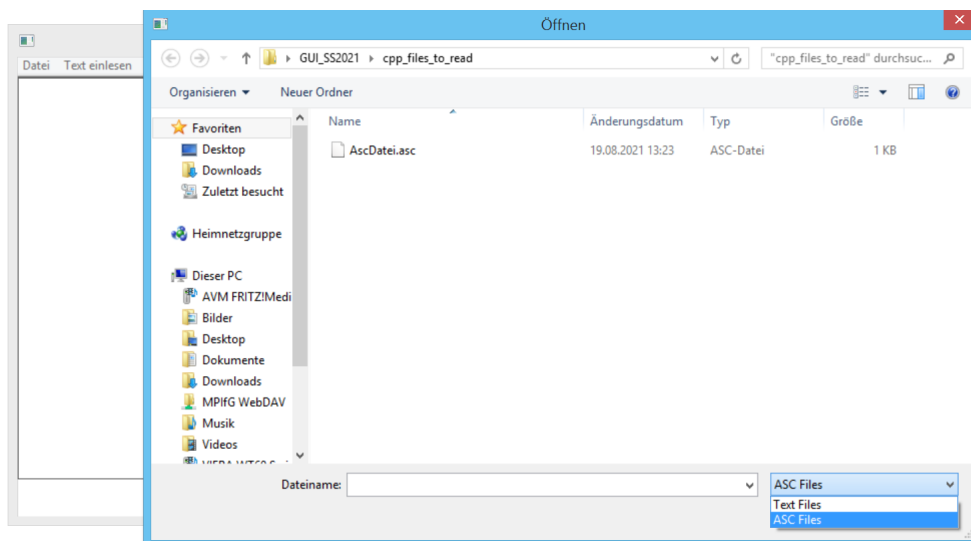


Abbildung 9: Auswählen einer .asc Datei

Wurde eine .txt oder .asc Datei ausgewählt, wird der Inhalt der Datei geladen und im Fenster dargestellt. Außerdem sieht man, dass sich der Titel des Fensters umgeändert hat, in den Namen der ausgewählten Datei.

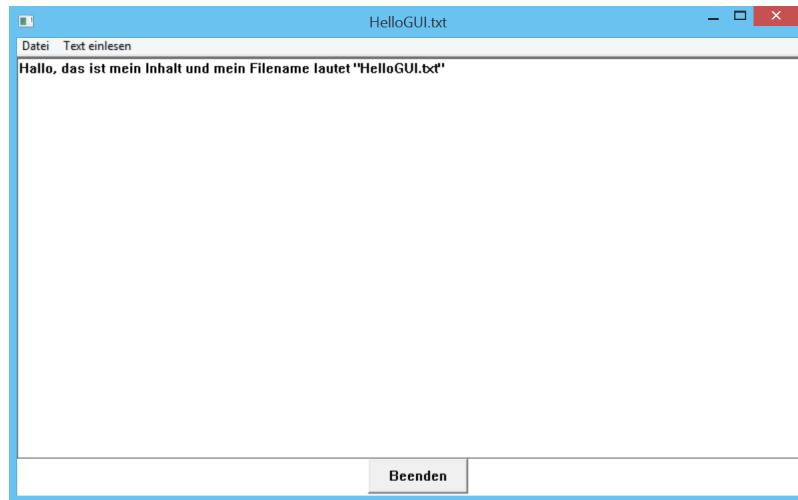


Abbildung 10: Geänderter Titel nach Auswahl der .txt Datei



Abbildung 11: Geänderter Titel nach Auswahl der .asc Datei

Der Text bleibt auch nach Verschieben, Vergrößern und Verkleinern des Fensters erhalten und wird weiterhin angezeigt.

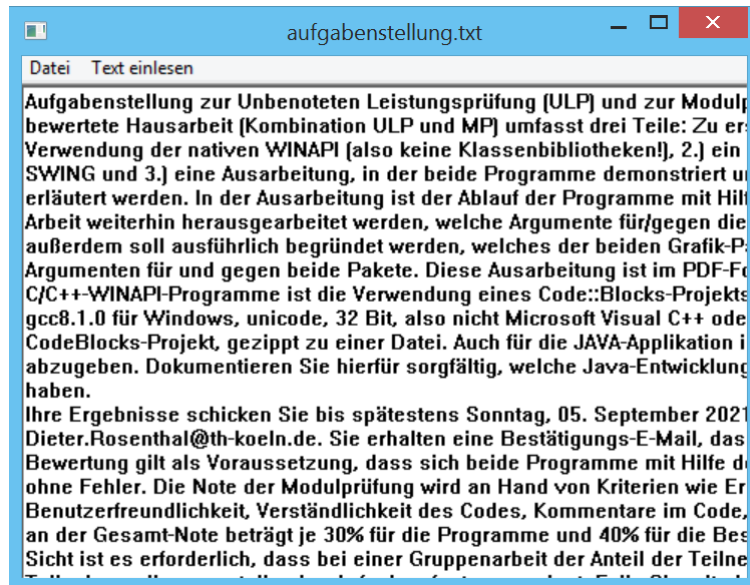


Abbildung 12: Fensteransicht, nach Verkleinerung

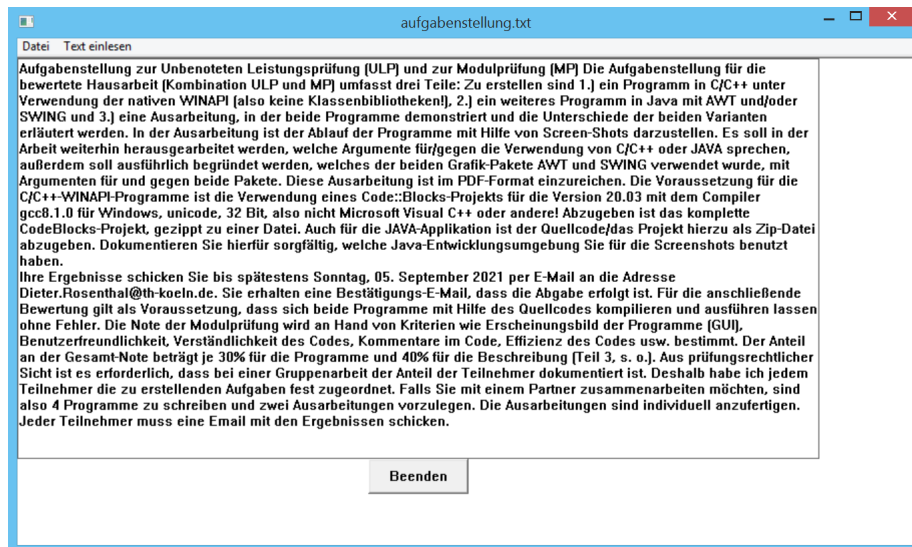


Abbildung 13: Fensteransicht, nach Vergrößerung

## 5 AWT oder Swing?

Zur Erstellung einer GUI mit Java gibt es verschiedene Optionen, die zur Auswahl stehen. In diesem Abschnitt wird jedoch auf AWT (Abstract Window Toolkit) und Java Swing eingegangen.

Für die Java Aufgabe in diesem Bericht, fiel die Wahl auf Swing. Die Gründe werden folgend erklärt, indem beide Optionen miteinander verglichen werden.

Einer der größten Unterschiede zwischen AWT und Swing ist, dass AWT abhängig ist vom Betriebssystem. Dies führt dazu, dass die Optik der Komponenten des Fenster, die des jeweiligen Betriebssystems übernehmen.

Auch erschwert dies, eine Anwendung ohne weiteres auf ein anderes System zu ziehen und dort ausführen zu lassen. Zudem führt dies auch zu einer starken Beanspruchung der Ressourcen des Betriebssystems.

Während Swing auf AWT aufbaut, läuft Swing unabhängig von der Plattform. Swing ist leichtgewichtiger als AWT und die Komponenten sind konfigurierbar. Dies ist nicht möglich bei AWT, da wie erwähnt dort die Komponenten des jeweiligen Betriebssystems verwendet werden.

Ein weiterer Vorteil, den die Swing Komponenten mit sich bringen ist, dass bei Swing mehr Komponenten zur Auswahl stehen, als bei AWT.

Während AWT größere Mengen an Speicher benötigt, kommt Swing mit weniger zurecht im Vergleich. Und in Sachen Geschwindigkeit ist Swing zusätzlich auch noch schneller als AWT.

Aus diesen Gründen wurde wie oben schon erwähnt für die Erstellung der Anwendung Swing verwendet.

## 6 Java oder C/C++ ?

Bei der Wahl zwischen C/C++ und Java als Sprache für die Umsetzung einer GUI Anwendung, gibt es verschiedene Punkte zu beachten.

Java ist eine objektorientierte Sprache, welche in einer Java Virtual Machine (JVM) ausgeführt wird. Dies ermöglicht es die Anwendung auf jedem System mit installierter JVM auszuführen und es Bedarf nur einer Kompilierung des Programms.

C++ ist eine Erweiterung der Programmiersprache C. Dies ermöglicht es, dass man mit ihr sowohl prozedural, als auch objektorientiert programmiert. Anders als bei Java, kann man den C/C++ Code nicht so einfach auf jedem System ausführen lassen. Hierfür ist es nötig, dass der Code auf dem entsprechenden System einmal zuvor kompiliert worden ist, bevor es darauf laufen kann. Außerdem sollte darauf geachtet werden, dass derselbe Compiler auch auf den weiteren Systemen zur Übersetzung des Programmcodes genutzt wird.

C/C++ ermöglicht den direkten Zugriff auf den Speicher des Systems und der Entwick-

ler ist zuständig für die Speicherverwaltung. Dies erlaubt es mehr Geschwindigkeit und Effizienz rauszuholen, wenn gut umgesetzt.

Java hingegen verwaltet den Speicher selbstständig im Hintergrund mit Hilfe eines Garbage Collector. Dafür ist es jedoch verglichen mit C/C++ langsamer.

C/C++ wird daher bevorzugt für maschinennahe Funktionalitäten verwendet, wie zum Beispiel Spieleentwicklung, Entwicklung von Hardware-Treibern oder Systementwicklungen, während Java sich im Bereich der Applikationen eingefunden hat, wie beispielsweise Web-, Enterprise- und Mobil-Applikationen.

In Java hat man auch eine breitere Auswahl an Komponenten, die man zur GUI Entwicklung verwenden kann. Und es gibt auch die Möglichkeit diese weiter anzupassen. Bei C/C++ zumindest mit der Nutzung der WINAPI wurde das Design des Systems übernommen.

Aus diesen Gründen lohnt es sich eher auf Java zurückzugreifen, wenn man eine Desktop Anwendung mit GUI Interface erstellen möchte, da diese unabhängig von der Plattform mit geringerem Aufwand eingesetzt werden kann, sofern Geschwindigkeit nicht an erster Position der Prioritäten steht.

## 7 Erklärung

Hiermit erkläre ich, Mahdi Bagheri, dass Bericht und Programme selbständig erarbeitet worden sind unter Verwendung der angegebenen Quellen.

## Literatur

- [1] <https://werner.rothschopf.net/modulo10.htm>
- [2] <https://planetcalc.com/2464/>
- [3] Java Swing GUI Tutorial (YouTube)
- [4] <https://www.java-tutorial.org/swing.html>
- [5] <https://docs.oracle.com/javase/tutorial/uiswing/>
- [6] C/C++ Win32 API Tutorials — Windows GUI Programming "windows.h" (YouTube)
- [7] Die WinAPI Plattform  
<http://www.win-api.de/tutorials.php>
- [8] theForger's Win32 API Programming Tutorial  
<http://www.winprog.org/tutorial/>
- [9] Programming reference for the Win32 API  
<https://docs.microsoft.com/en-us/windows/win32/api/>
- [10] <https://newbedev.com/c-how-to-get-filename-from-path-in-c-code-example>
- [11] <https://www.javatpoint.com/awt-and-swing-in-java>
- [12] <https://www.javatpoint.com/cpp-vs-java>
- [13] A Comparison of Programming Languages for Graphical User Interface Programming  
[shorturl.at/otEN5](http://shorturl.at/otEN5)