**Dokumentation IDPA Vorprojekt**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Namen der** | **Gruppenmitglieder** |  |
| Shane Röllin | Andrea Niklaus | Nathan Göhl | David Zumstein |

**Klasse:** I3b/IM21d

**Projektname:** SAND

**Zuständige Lehrpersonen:** Caroline Streif, Michael Schneider

**Datum Beginn:** 16.08.2023

**Datum Ende:** 27.09.2023

**Inhaltsverzeichnis**

Inhalt

[**Teil 1:** 4](#_Toc146745712)

[**1.0** Disposition/Projektvertrag 4](#_Toc146745713)

[**2.0** Deklaration der Vorkenntnisse 5](#_Toc146745714)

[**3.0** Deklaration der Vorarbeiten 5](#_Toc146745715)

[**4.0** Deklaration der benützten Firmenstandards 5](#_Toc146745716)

[**5.0** Zeitplan 5](#_Toc146745717)

[**6.0** Arbeitsjournal 5](#_Toc146745718)

[**6.1** Arbeitsjournal 16.08.2023 5](#_Toc146745719)

[**6.2** Arbeitsjournal 23.08.2023 6](#_Toc146745720)

[6.3 Arbeitsjournal 30.08.2023 7](#_Toc146745721)

[6.4 Arbeitsjournal 06.09.2023 7](#_Toc146745722)

[6.5 Arbeitsjournal 13.09.2023 8](#_Toc146745723)

[6.6 Arbeitsjournal 20.09.2023 8](#_Toc146745724)

[7 Organisation der Arbeitsergebnisse 9](#_Toc146745725)

[8 Anleitung zur Installation und zur Bedienung des Produkts 9](#_Toc146745726)

[**Teil 2:** 11](#_Toc146745727)

[**Management Summary:** 11](#_Toc146745728)

[Ausgangslage: 11](#_Toc146745729)

[Vorgehen: 11](#_Toc146745730)

[Ergebnis: 11](#_Toc146745731)

[**1.1** Beschreibung der Informationsphase mit Vorgehen, Informationsquellen, Problemen Informationsquelle: 11](#_Toc146745732)

[Vorgehen: 11](#_Toc146745733)

[Informationsquelle: 11](#_Toc146745734)

[**2.1** Tätigkeitsliste mit einer verantwortlichen Person und einem Erfüllungsdatum pro Tätigkeit. Dabei sollen die geplanten und die tatsächlich ausgeführten Tätigkeiten ersichtlich sein. 12](#_Toc146745735)

[**3.1** Architektur des Programmes und der Daten: Überblick, Beschreibung und fachlich korrekte Begründung 13](#_Toc146745736)

[**4.1** GUI-Prototypen mit Usability-Überlegungen 13](#_Toc146745737)

[**5.1** Klar dokumentierte Entscheidungen 14](#_Toc146745738)

[**6.0** Projektbeschreibung 14](#_Toc146745739)

[6.2 Lösung beschreiben und erklären: Umfeld 15](#_Toc146745740)

[6.3 Präzisierung der Aufgabenstellung 15](#_Toc146745741)

[6.4 Abklärungen 15](#_Toc146745742)

[6.5 Lösungsvarianten 15](#_Toc146745743)

[6.6 Auswahlkriterien 15](#_Toc146745744)

[6.7 Realisierung: 15](#_Toc146745745)

[6.8 Probleme und Lösungen 15](#_Toc146745746)

[**7.1**  Schlüsselstellen von Programmen mit Erklärungen 15](#_Toc146745747)

[**8.1** Beschreibung des tatsächlichen GUI 16](#_Toc146745748)

[**9.1** Testfälle, Testprotokolle und Testberichte 16](#_Toc146745749)

[**Testfälle:** 16](#_Toc146745750)

[**Testprotokoll:** 17](#_Toc146745751)

[**Testbericht:** 17](#_Toc146745752)

[**10.1** Abgrenzung und klare Deklaration der Eigenleistungen und Protokolle aller KI-Sitzungen. 17](#_Toc146745753)

[**11.1** Sitzungs-Protokolle und Resultate 18](#_Toc146745754)

[**Schlusswort:** 18](#_Toc146745755)

# **Teil 1:**

1. Disposition/Projektvertrag

|  |  |
| --- | --- |
| Projektname: | SAND |
| GruppenmitgliederInnen  mit Klassen und Emailadresse: | IM21d/I3b:  Shane Röllin: [shane.roellin@students.ksba.ch](mailto:shane.roellin@students.ksba.ch)  Andrea Niklaus: [andrea.niklaus@students.ksba.ch](mailto:andrea.niklaus@students.ksba.ch)  Nathan Göhl: [nathan.goehl@students.ksba.ch](mailto:nathan.goehl@students.ksba.ch)  David Zumstein: [david.zumstein@students.ksba.ch](mailto:david.zumstein@students.ksba.ch) |
| Betreuende Lehrpersonen mit Emailadresse: | Caroline Streif Schmid : [caroline.streif-schmid@kanti-baden.ch](mailto:caroline.streif-schmid@kanti-baden.ch)  Michael Schneider: [michael.schneider@kanti-baden.ch](mailto:michael.schneider@kanti-baden.ch) |
| Titel: | IDPA Vorprojekt |
| Beschreibung: | Dies ist das Vorprojekt des IDPA |
| Startdatum des Vorprojekts: | 16.08.23 |
| Abgabe mit Zeitpunkt: | 27.09.23 um 23:59 Uhr |
| Anforderungen: | Im Projekt beschrieben |
| Datum und Unterschrift  GruppenmitgliederInnen: | 27.09.23: |
| Datum und Unterschrift betreuende  Lehrpersonen: | - |

1. Deklaration der Vorkenntnisse

Modul 320: Wir haben im Modul 320 Objektorientiertes Programmieren kennengelernt. Dort haben wir C# benutzt.

Modul 426: In diesem Modul haben wir uns mehr in Scrum fokussiert und allgemein Git.

Modul 306: In diesem Modul haben wir gelernt, wie ‘richtig’ zu dokumentieren und auf was wir achten müssen. Auch haben wir uns mehr auf IPERKA in diesem Modul fokussiert.

Neuland ist vom Programmieren her nicht mehr so vieles, ausser den Zinsrechnen-Part.

1. Deklaration der Vorarbeiten

keine

1. Deklaration der benützten Firmenstandards

keine

1. Zeitplan

2 Lektionen Blöcke:

|  |  |  |
| --- | --- | --- |
| **Tag (Soll)** | **Was?** | **Ist** |
| 16.08.23 | Projektplanung | 16.08 |
| 23.08.23 | Projektplanung | Programmieren | Arbeitsjournal | 23.08 |
| 30.08.23 | Programmieren | Dokumentation | Arbeitsjournal | 30.08 |
| 06.09.23 | Programmieren | Dokumentation | Arbeitsjournal | 06.09 |
| 13.09.23 | Programmieren | Dokumentation | Arbeitsjournal | 13.09 |
| 20.09.23 | Programmieren | Dokumentation | Arbeitsjournal | 20.09 |

1. Arbeitsjournal
   1. Arbeitsjournal 16.08.2023

|  |  |  |
| --- | --- | --- |
| Datum | 16.08.2023 |  |
| Ausgeführte Tätigkeiten: | Tätigkeit | Person |
|  | Grundrüstung Programm erstellt | Nathan Göhl |
|  | Aufgabenstellung deutlicher geschrieben | Andrea Niklaus, David Zumstein, Shane Röllin |
| Erreichte Ziele | Ziel: | Person: |
|  | Aufgabenstellung gelesen | Andrea Niklaus, David Zumstein, Shane Röllin, Nathan Göhl |
|  | Mindmap/Zusammenfassung erstellt | Andrea Niklaus |
| Aufgetretene Probleme: | Aufgabenstellung zuerst unklar gewesen |  |
| Hilfestellung (wer, was): | Lehrperson (hat sie überarbeitet |  |
| Überzeit | Keine |  |
| Reflexion | Die Arbeit im Team lief gut, wir konnten jedoch nicht richtig mit dem Projekt anfangen, da die Aufgabenstellung zu diesem Zeitpunkt nicht klar war. |  |

* 1. Arbeitsjournal 23.08.2023

|  |  |  |
| --- | --- | --- |
| 1. Datum | 22.08.2023 |  |
| Ausgeführte Tätigkeiten: | Tätigkeit | Person |
|  | Programm erarbeitet | Andrea Niklaus, David Zumstein, Nathan Göhl, Shane Röllin |
|  | Dokumentation geschrieben | Andrea Niklaus, David Zumstein, Nathan Göhl, Shane Röllin |
|  | Arbeitsjournal geschrieben | Andrea Niklaus |
| Erreichte Ziele | Ziel: | Person: |
|  | Aufgabenstellung neu geschrieben/verbessert | Andrea Niklaus |
|  | Zinsberechnung funktioniert | Andrea Niklaus, David Zumstein, Nathan Göhl, Shane Röllin |
| Aufgetretene Probleme: | Gruppenaufteilung | Andrea Niklaus |
| Hilfestellung (wer, was): | Gruppe selbst gelöst |  |
| Überzeit | Keine |  |
| Reflexion | Die Gruppenaufteilung war trotz nach 2 Wochen arbeiten nicht richtig klar, jedoch konnten wir dies lösen. | Andrea Niklaus |

6.3 Arbeitsjournal 30.08.2023

|  |  |  |
| --- | --- | --- |
| Datum | 30.08.2023 |  |
| Ausgeführte Tätigkeiten: | Tätigkeit | Person |
|  | Programm erarbeitet | Andrea Niklaus, David Zumstein, Nathan Göhl, Shane Röllin |
|  | Dokumentation geschrieben | Andrea Niklaus, David Zumstein, Nathan Göhl, Shane Röllin |
|  | Arbeitsjournal geschrieben | Andrea Niklaus |
| Erreichte Ziele | Ziel: | Person: |
|  | Dokumentation weiterbearbeitet | Andrea Niklaus, David Zumstein, Nathan Göhl, Shane Röllin |
|  | Zinsberechnung funktioniert | Andrea Niklaus, David Zumstein, Nathan Göhl, Shane Röllin |
| Aufgetretene Probleme: | Keine |  |
| Hilfestellung (wer, was): | - |  |
| Überzeit | Keine |  |
| Reflexion | Die Gruppenarbeit lief gut, wir sind gut im Zeitplan. | Andrea Niklaus |

* 1. Arbeitsjournal 06.09.2023

|  |  |  |
| --- | --- | --- |
| 1. Datum: | 06.09.2023 | |
| Ausgeführte Tätigkeiten: | Tätigkeit: | Person: |
| Programm erarbeitet | Röllin, Göhl, Niklaus, Zumstein |
| Dokumentation weitergeführt | Niklaus, Zumstein |
| Erreichte Ziele: | Ziel: | Person: |
| Programmierungszeitplan eingehalten | Röllin, Göhl, Niklaus, Zumstein |
| Dokumentation verbessert | Niklaus, Zumstein |
| Aufgetretene Probleme: | Problem: | Person: |
| Mergen vom Code bei GitHub Desktop | Göhl |
| - | - |
| Hilfestellung (wer, was): | - | |
| Pendenzenliste | Pendenz: | Person: |
| Programm fertig machen | Röllin, Göhl, Niklaus, Zumstein |
| Arbeitsjournal erstellen | Zumstein |
| Reflexion: | Reflexion: | Person: |
| Die Letzte Arbeitssession war gut, wir haben effizient gearbeitet ohne grosse Stolpersteine. Wir haben auch gemerkt, dass wir Online genau so gut wie vor Ort zusammen arbeiten können. | Zumstein |
| - | - |
| Vergleich mit dem Zeitplan: | Vergleich mit Zeitplan: | Person |
| Wir stehen gerade perfekt im Zeitplan und haben genau so viel erreicht, wie wir geplant haben | Zumstein |
| - | - |

* 1. Arbeitsjournal 13.09.2023

Kein Arbeitsjournal abgegeben

* 1. Arbeitsjournal 20.09.2023

|  |  |  |
| --- | --- | --- |
| Datum: | 20.09.2023 | |
| Ausgeführte Tätigkeiten: | Tätigkeit: | Person: |
| Dokumentation weitergeführt | Röllin, Göhl, Niklaus, Zumstein |
| - | - |
| Erreichte Ziele: | Ziel: | Person: |
| ¾ der Dokumentation erarbeitet | Röllin, Göhl, Niklaus, Zumstein |
| - | - |
| Aufgetretene Probleme: | Problem: | Person: |
| - | - |
| - | - |
| Hilfestellung (wer, was): | - | |
| Pendenzenliste | Pendenz: | Person: |
| Dokumentation fertigstellen | Röllin, Göhl, Niklaus, Zumstein |
| Arbeitsjournal erstellen | Zumstein |
| Reflexion: | Reflexion: | Person: |
| Wir hatten unser Programm schon fertig und konnten uns auf die Dokumentation konzentrieren. Wir haben nun momentan so ¾ der Dokumentation. Lief ganz gut | Zumstein |
| - | - |
| Vergleich mit dem Zeitplan: | Vergleich mit Zeitplan: | Person |
| Gut im Zeitplan, müssen nur noch Doku fertig machen und dann können wir noch Anpassungen machen, wenn wir das so wollen | Zumstein |
| - | - |

1. Organisation der Arbeitsergebnisse

Alle Dateien sind auf <https://github.com/IronEsther/IDPA_Vorprojekt> hochgeladen. Dort herrscht auch eine Ordner-Hierarchie, das Projekt an sich und die Dokumentation sind je in einem separaten Ordner gespeichert.

1. Anleitung zur Installation und zur Bedienung des Produkts

Laden Sie den Ortner [IDPA\_Vorprojekt\_SAND](https://github.com/IronEsther/IDPA_Vorprojekt/tree/main/IDPA_Vorprojekt_SAND) aus unserem GitHub herunter.

Öffnen Sie Visual Studio 2022.

Öffnen Sie den Ordner. Ein Bild, das Text, Screenshot, Schrift, Design enthält.

Automatisch generierte Beschreibung

Suchen Sie den Ordner aus, den Sie sich vorher heruntergeladen haben.

Schauen Sie, dass Sie das File Program.cs als File ausgewählt haben, welches sie starten. 

Starten Sie nun das Program, indem Sie folgenden Knopf drücken: 

Befolgen Sie nun die Anweisungen des Programs.

# **Teil 2:**

## **Management Summary:**

### Ausgangslage:

Wir mussten für die fiktive Bank ‘Goldstein und Klaster Bank’ den individuellen Marchzins-Bonus (siehe Ausgangslage vom Arbeitsauftrag 1001) erweitern und den Prozess erleichtern und effizienter gestalten. Unsere Lehrer waren die ‘Leiter’ und ‘Kundenberater’, welche uns bei diesem Prozess begleiteten und uns ihre Wünsche mitteilten.

### Vorgehen:

Wir haben dieses Projekt in den letzten 6 Wochen bearbeitet. Wir haben das Projekt hauptsächlich Online durchgeführt, wobei auch unsere IDPA-Lektionen per Remote-Teaching waren.

### Ergebnis:

Das Resultat des Projektes sieht positiv aus. Wir brauchen nun noch die Meinung von unseren Lehrern. Bis zum Schluss haben wir unser Bestes gegeben und freuen uns auf die Rückmeldung.

## **1.1** Beschreibung der Informationsphase mit Vorgehen, Informationsquellen, Problemen Informationsquelle:

### Vorgehen:

Die Lehrer haben uns das Projekt vorgestellt. Wir haben analysiert, wie wir dieses Projekt so gut wie möglich umsetzten können und haben bei Fragen diese den Lehrern gestellt.

### Informationsquelle:

Worddokument Zinsberechnung Arbeitsauftrag 1001:

Lern- und Arbeitsauftrag 1001

|  |  |
| --- | --- |
| **Titel:** | **Zinsberechnung** |
| **Modul:** | IDPA Informatiker/in EFZ |
| **Autor / Version:** | Michael Schneider, Brigitte Marti, Caroline Streif Schmid |
| **Hilfsmittel:** | Alles |

Ausgangslage

Sie arbeiten mit anderen Praktikantinnen und Praktikanten bei der "Goldstein und Klaster Bank", einem modernen Kreditinstitut, das individuelle Spar- und Anlageprodukte für seine Kunden entwickelt. In den letzten Jahren hat die Bank den "Marchzins-Bonus" eingeführt, bei dem Kunden für Ihren Geburtstag im jeweiligen Monat einen besonderen Zins (1. bis zum Geburtstag) auf ihre Sparbeträge erhalten können. Dieser Bonus dient als Anreiz für Bestandskunden, ihre Geldanlagen zu erhöhen, und als Lockmittel für potenzielle Neukunden.

Die Filialleiter und Kundenberater müssen oft den zusätzlichen Zinsbetrag, den ein Kunde durch den Bonus verdienen kann, schnell berechnen. Sie möchten dieses Prozedere erleichtern und effizienter gestalten.

Aufgabe:

Sie wurden beauftragt, ein Tool zu erstellen, das den zusätzlichen Zinsbetrag berechnet, den ein Kunde durch den Bonus auf seine Sparbeträge erhalten kann. Dabei soll der zu erhaltene Betrag und die Verrechnungssteuer getrennt ausgewiesen werden.

Designen und realisieren Sie dieses Programm inklusive Dokumentation.

Hinweise:

Es handelt sich um einen Marchzins, der berechnet werden muss.

Sie sind zuständig dafür, die Anforderungen zu erstellen. Ihre Lehrpersonen dienen dabei als Auftraggeberinnen und Auftraggeber.

## **2.1** Tätigkeitsliste mit einer verantwortlichen Person und einem Erfüllungsdatum pro Tätigkeit. Dabei sollen die geplanten und die tatsächlich ausgeführten Tätigkeiten ersichtlich sein.

**Tätigkeitsliste:**

**Durchgestrichene: ~~geplant aber nicht gemacht~~**

|  |  |  |
| --- | --- | --- |
| Tätigkeit | Erfüllungsdatum | Person |
| Arbeitsjournal erstellen | 17.08.2023 | Niklaus |
| Erste Dokumentationsschritte machen | 16.08.2023 | Niklaus |
| Entscheidungen bezüglich Entwicklung treffen | 16.08.2023 | Göhl |
| Projekt und Repo Mal erstellen | 16.08.2023 | Göhl |
| Zinsformel berechnen | 16.08.2023 | Zumstein |
| Arbeitsjournal2 erstellen | 24.08.2023 | Niklaus |
| Zinsformel in VS erstellen | 23.08.2023 | Röllin |
| Arbeitsjournal3 erstellen | 31.08.2023 | Zumstein |
| Abfrage erstellen | 30.08.2023 | Niklaus |
| Programm Ablauf erstellen | 30.08.2023 | Röllin |
| Arbeitsjournal4 erstellen | 07.09.2023 | Zumstein |
| Clean code | 06.09.2023 | Röllin |
| Hauptlogik erstellen | 06.09.2023 | Göhl |
| Fehler Behandlung erstellen | 06.09.2023 | Röllin |
| ~~Arbeitsjournal5 erstellen~~ | ~~14.09.2023~~ | ~~Zumstein~~ |
| Fehler Behandlung2 erstellen | 13.09.2023 | Zumstein |
| weitere Kundenabfrage erstellt | 13.09.2023 | Göhl |
| Programm überprüft | 13.09.2023 | Göhl |
| Arbeitsjournal6 | 21.09.2023 | Zumstein |
| Dokumentation Updaten | 20.09.2023 | Niklaus |
| Dokumentation abschliessen | 27.09.2023 | Niklaus |

## **3.1** Architektur des Programmes und der Daten: Überblick, Beschreibung und fachlich korrekte Begründung

Namespace und Klassenstruktur: Das Programm verwendet den Namespace IDPA\_Vorprojekt\_SAND und enthält die Hauptklasse Program. Innerhalb dieser Klasse gibt es verschiedene Methoden, um die Abfrage von Benutzereingaben, die Berechnung von Zinserträgen und die Verwaltung des Programmablaufs durchzuführen.

Hauptmethode (Main): Die Main-Methode dient als Einstiegspunkt des Programms und enthält eine Endlosschleife (while (true)), die solange läuft, bis das Programm explizit beendet wird. In dieser Schleife werden die Methoden Abfrage(), ZinsertragFormel() und WeitererKunde() nacheinander aufgerufen.

Abfragemethode (Abfrage()): Diese Methode wird verwendet, um Benutzereingaben für Kapital, Geburtstag und Zinssatz entgegenzunehmen. Sie enthält Aufrufe von Methoden wie Tastatur() und Eingabe() zur Interaktion mit dem Benutzer und zur Validierung der Eingaben.

Eingabe-Methode (Eingabe(int Funktion)): Diese Methode behandelt Benutzereingaben und gibt einen double-Wert zurück. Sie enthält Validierungslogik, um sicherzustellen, dass die Eingaben gültig sind. Abhängig von der übergebenen Funktion werden spezielle Validierungen durchgeführt, z.B. für das Geburtsdatum.

Variablen für die Zinsberechnung: Das Programm verwendet verschiedene öffentliche double-Variablen wie Kapital, Geburtstag, Zinssatz und Bonus, um Werte für die Zinsberechnung zu speichern. Diese Variablen werden von verschiedenen Methoden verwendet.

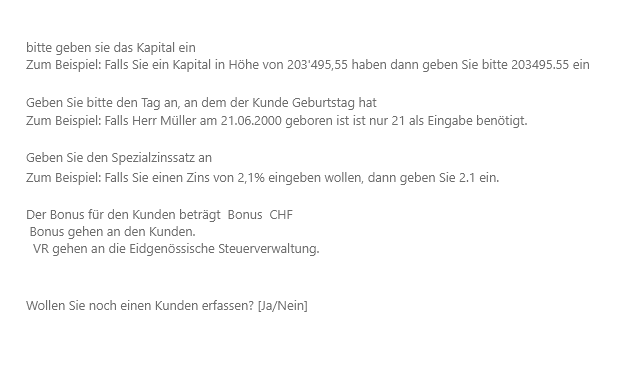
Zinsertrag-Berechnungsmethode (ZinsertragFormel()): Diese Methode berechnet den Zinsertrag für den Kunden anhand der eingegebenen Daten. Sie verwendet die gespeicherten Variablen und gibt die Ergebnisse in der Konsole aus.

Weiterer Kunde-Methode (WeitererKunde()): Diese Methode fragt den Benutzer, ob ein weiterer Kunde erfasst werden soll, und beendet das Programm, wenn die Antwort "Nein" ist.

Fehlerbehandlung: Das Programm enthält eine einfache Fehlerbehandlung, die auftritt, wenn der Benutzer ungültige Eingaben macht. Sie löscht die vorherige Konsolenausgabe und gibt eine Fehlermeldung aus.

Die Architektur dieses Programms folgt einem einfachen Konsolenanwendungsdesign, bei dem Benutzereingaben erfasst, verarbeitet und auf Fehler überprüft werden. Die Trennung der Funktionalität in verschiedene Methoden macht den Code wartbarer und übersichtlicher. Es wäre jedoch ratsam, weitere Kommentare und Dokumentation hinzuzufügen, um den Code für andere Entwickler verständlicher zu machen und die Wartbarkeit zu verbessern.

## **4.1** GUI-Prototypen mit Usability-Überlegungen



Wir wollen es dem Benutzer möglichst einfach machen, weswegen wir oftmals ein Beispiel zusätzlich angegeben haben

## **5.1** Klar dokumentierte Entscheidungen

* Eine wichtige Entscheidung in unserem Projekt war, in welchem Umfeld wir es umsetzten. Wir haben uns schlussendlich auf die Konsole-Applikation entschieden, da wir das Projekt somit besser umsetzen konnten.
* Entscheid, ‘Text-Typer’ benutzen: Um unser Projekt einzigartig zu machen, haben wir uns entschieden, einen Code-Abschnitt aus dem Internet zu benutzen, welcher macht, dass die Konsole-Ausgabe besser dargestellt wird.

## **6.0** Projektbeschreibung

6.1 Anforderungen (von uns hinzugefügt)

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Muss / Kann?** | **Funktional? Qualität? Rand?** | **Beschreibung** |
| 1 | Muss | Rand | Das Programm ist in C# geschrieben |
| 2 | Muss | Rand | Das Programm ist als eine Konsolenanwendung in Visual Studio erstellt. |
| 3 | Muss | Funktional | Zins kann berechnet werden. |
| 4 | Muss | Funktional | Erhaltener Betrag und Verrechnungssteuer werden separat ausgewiesen. |
| 5 | Muss | Funktional | Man kann den Zinssatz selbst bestimmen |
| 6 | Muss | Funktional | Man kann das Geburtsdatum des Kunden eingeben |
| 7 | Muss | Funktional | Man kann das Kapital des Kunden eingeben |
| 8 | Kann | Funktional | Man kann den Zins für mehrere Kunden mit einmal Programm ausführen berechnen |
| 9 | Kann | Qualität | Der code ist clean. |

6.2 Lösung beschreiben und erklären: Umfeld

Die einzigen Schnittstellen, die es in unserem Programm gibt, sind die Benutzerschnittstellen. Dort wertet das Programm die Benutzereingaben aus und gibt je nach Verarbeitung eine andere Ausgabe.

6.3 Präzisierung der Aufgabenstellung

Es wurde von uns gefordert, eine Applikation zu erstellen, welche einen speziellen ‘Geburtstag-Marchzins’ aus den Benutzereingaben berechnet. Unsere Kunden wollten damit ihre Effizienz steigern. Mehr Angaben haben wir vom Kunden nicht.

6.4 Abklärungen

Wir hatten keine spezifische Abklärung mit den Kunden

6.5 Lösungsvarianten

Unsere einzige Lösungsvariante ist auf [IDPA-Vorprojekt](https://github.com/IronEsther/IDPA_Vorprojekt) hochgeladen.

6.6 Auswahlkriterien

Es gab keine spezifischen Auswahlkriterien, welche wir erfüllen mussten.

6.7 Realisierung:

|  |  |  |
| --- | --- | --- |
| Tätigkeit | Erfüllungsdatum | Person |
| Projekt und Repo erstellen | 16.08.2023 | Göhl |
| Zinsformel berechnen | 16.08.2023 | Zumstein |
| Zinsformel in VS erstellen | 23.08.2023 | Röllin |
| Abfrage erstellen | 30.08.2023 | Niklaus |
| Programm Ablauf erstellen | 30.08.2023 | Röllin |
| Clean Code | 06.09.2023 | Röllin |
| Hauptlogik erstellen | 06.09.2023 | Göhl |
| Fehler Behandlung erstellen | 06.09.2023 | Röllin |
| Fehler Behandlung2 erstellen | 13.09.2023 | Zumstein |
| weitere Kundenabfrage erstellt | 13.09.2023 | Göhl |
| Programm überprüft | 13.09.2023 | Göhl |

Wir haben das Programm in der Konsolenapplikation mit der Sprache C# in Visual Studio 2022 realisiert. Um den Code zu teilen haben wir GitHub benutzt, da jeder in der Gruppe damit vertraut ist.

6.8 Probleme und Lösungen

Es gab generell fast keine Probleme, jedoch gab es einen Konflikt bei der Projektdokumentation, in dem fast alles gelöscht wurde. GitHub konnte unsere Dokumentation nicht mergen, da mehrere Leute gleichzeitig etwas geändert hatten Wir konnten es zum Glück jedoch beheben. Wir haben es behoben, indem Andrea Niklaus das Dokument mit allen geteilt hat und wir einfach alle Manuell unsere Veränderungen reingeschrieben haben.

## **7.1** Schlüsselstellen von Programmen mit Erklärungen

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

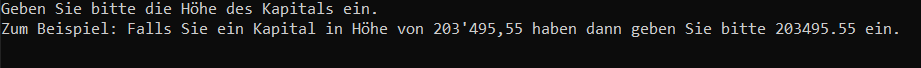
Hiermit berechnen wir den Bonus den der Kunde erhält nachdem alles eigegeben ist.  
Und wir berechnen die Verrechnungssteuer.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Hier geben wir die Anweisungen an den Benutzer und sagen im was zu tun ist. Wir haben es so gemacht, dass die Anweisung so auftaucht, als würde es von jemandem geschrieben werden und danach eine kurze Erklärung sofort und ohne schreib Animation auftaucht, die dem benutzer erklärt in welchem Format er seinen Input machen soll.

## **8.1** Beschreibung des tatsächlichen GUI

Es wird jeweils ein Prompt ausgegeben, welcher dem Benutzer eine Anweisung gibt.  


Wenn der Benutzer etwas falsches eingibt, kommt eine Fehlermeldung und dann muss der Benutzer eine neue Eingabe machen. 

Wenn eine korrekte Eingabe getätigt wird gibt das Program eine neue Anweisung. Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Wenn man alle Eingaben angegeben hat wird der Bonus und die Verrechnungssteuer ausgegeben. Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Nun kann man entscheiden ob man einen weiteren Kunden erfassen will. Wenn man «Ja» angibt, wird der ganze Ablauf noch mal von Anfang an gestartet. Wenn man «Nein» angibt schliesst sich das Program.

## **9.1** Testfälle, Testprotokolle und Testberichte

### **Testfälle:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Voraussetzung** | **Eingabe** | **Erwartete Ausgabe** |
| 1.1 | Programm gestartet | Zinssatz, Geburtsdatum, Kapital | Berechneter Zins |
| 1.2 | Zins berechnet | - | Erhaltener Betrag und Verrechnungssteuer werden separat ausgewiesen |
| 3.1 | Programm gestartet | Kapital, Geburtsdatum | "bitte geben sie den Spezialzinssatz ein" |
| 4.1 | Programm gestartet | Kapital | "Geben Sie bitte den Tag an, an dem der Kunde Geburtstag hat" |
| 5.1 | Programm gestartet | - | "Geben Sie bitte die Höhe des Kapitals an" |
| 6.1 | Zins berechnet  Spiel gespielt | -  Enter-Taste | Programm schliesst sich  "Wollen sie noch einen Kunden erfassen?" |

### **Testprotokoll:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Datum** | **Resultat** | **Durchgeführt** |
| 1.1 | 20.09.2023 | OK | Göhl |
| 1.2 | 20.092023 | OK | Göhl |
| 3.1 | 20.09.2023 | OK | Göhl |
| 4.1 | 20.09.2023 | OK | Göhl |
| 5.1 | 21.09.2023 | OK | Göhl |
| 6.1 | 21.09.2023 | OK | Göhl |

### **Testbericht:**

Unser Programm erfüllt die Anforderungen und hat alle Testfälle ohne Probleme bestanden. Das Programm ist bereit, um an den Kunden übergeben zu werden.

## **10.1** Abgrenzung und klare Deklaration der Eigenleistungen und Protokolle aller KI-Sitzungen.

public static void FehlerBehandlung()

{

/\* Fehlerbehandlung für Eingabe.

Der Code ändert die letzte Zeile in der Konsole um die Konsolenausgabe besser aussehen zu lassen.

Quelle: <https://stackoverflow.com/a/8946847> \*/

int currentLineCursor = Console.CursorTop;

Console.SetCursorPosition(0, Console.CursorTop - 1);

Console.Write(new string(' ', Console.WindowWidth));

Console.SetCursorPosition(0, Console.CursorTop);

Für 3.1 haben wir nur mit Chat GPT gearbeitet.

Prompt:  
Wir haben den ganzen Program Code eingefügt und diese Zeilen zusätzlich dazu geschrieben:

«Gib mir eine Antwort zu diesem Punkt:

Architektur des Programmes und der Daten: Überblick, Beschreibung und fachlich korekte Begründung»

Chat GPT hat uns dann den Output gegeben und wir haben diesen eins zu eins übernommen.

* 1. Sitzungs-Protokolle und Resultate

Sitzungs-Protokolle:

* 16.08.23: Besprechung Projektumsetzung und Auftrag
* 23.08.23: Zuteilung -> wer was macht + Zeitplan, Anforderungen und Tätigkeitsliste erstellt

# **Reflexion:**

Die IDPA-Arbeit an sich lief gut, jedoch gab es schon am Anfang Stolpersteine, da die Aufgabenstellung nicht klar genug war. Die Lehrpersonen haben die Aufgabenstellung überarbeitet. Sonst lief das Vorprojekt sehr gut, die Gruppenarbeit und die Arbeitsaufteilung lief wie geplant und auch das Projekt und die Dokumentation waren am Schluss fertig. Ein Verbesserungsvorschlag für das nächste Projekt wäre, dass wir den Lehrpersonen mehr Fragen stellen sollten, wie z.B. wie sie die Applikation am Schluss haben möchten, wie das Layout sein sollte, etc. Da sie ja die ‘Kunden’ sind und wir sie mit diesem Projekt/mit dieser Abgabe befriedigen wollten.

# **Schlusswort:**

**Persönliches Fazit aller am Projekt beteiligten:**

* *Unser Projekt ist funktionsfähig, was ich sehr toll finde, was ich nicht so toll finde ist aber die ewig lange Dokumentation* ***– Zumstein David***
* *Die Implementierung des Projekts an sich war eine spannende und eine informative Sache, jedoch schliesse ich mich Herrn Zumstein wegen der Dokumentation an.* ***– Niklaus Andrea***
* *Wir haben es sehr schnell geschaft das Program zum laufen zu bringen. Doch die Dokumentation war mit Abstand die anstrengenste Arbeit.* ***– Göhl Nathan***
* *Das Programm, zu überarbeiten und zu verbessern hat mir an diesem Projekt sehr viel Spass gemacht, die Dokumentation hat sich jedoch ordentlich gezogen.* ***– Shane Röllin***

**Quellenverzeichnis:**

<https://stackoverflow.com/a/8946847>

**Glossar:** keine IDPA-spezifische Begriffe benutzt.