Lernatelier: Projektdokumentation

Eslem Akdemir, Elisa Petit-Sirigu

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Version** | **Änderung** | **Autor** |
| 11.01.2024 | 0.0.1 | Informieren, Planen, Entscheiden | Eslem Akdemir, Elisa Petit-Sirigu |
| 18.01.2024 | 0.0.2 | Quellen, Planen | Elisa Petit-Sirigu |
| 25.01.2024 | 0.0.3 | Informieren | Eslem Akdemir, Elisa Petit-Sirigu |
| 01.02.2024  08.02.2024 | FERIEN | | |
| 29.02.2024 | 0.0.4 | Realisieren, Kontrollieren | Eslem Akdemir, Elisa Petit-Sirigu |
| 07.03.2024 | 1.0.0 | Auswerten | Eslem Akdemir, Elisa Petit-Sirigu |

# Informieren

## Ihr Projekt

Das Projekt ist eine individuelle Arbeit. Wir werden beide unsere OOP-Kenntnisse verbessern und üben objektorientiert zu programmieren. Die Zusammenarbeit besteht in der Unterstützung. Wir werden uns gegenseitig helfen und unterstützen. Allerdings werden wir an unterschiedlichem Code arbeiten und das auch beides auf GitHub veröffentlichen. Somit kann man unsere Entwicklung verfolgen.

Elisa Petit-Sirigu  
Es ist ein Spiel mit 3 Würfeln. Sie können es mit mehreren Personen spielen. Am Anfang hat jeder Spieler 3 Chips und kann 3 Würfel werfen. Ein Spieler kann nur alle 3 Würfel werfen, wenn er/sie mindestens 3 Chips hat. Hat der Spieler einen oder zwei Chips, muss er/sie zwei Würfel werfen. Jeder Würfel bestimmt, was mit einem Chip passiert. Wenn du eine Eins würfelst, passiert nichts mit dem Chip. Bei einer Zwei passiert auch nichts mit dem Chip. Bei einer Drei passiert ebenfalls nichts mit dem Chip. Bei einer Vier musst du einen Chip nach links geben. Bei einer Fünf musst du einen Chip in die Mitte werfen. Bei einer Sechs musst du einen Chip nach rechts geben.

Nachdem du die Würfel geworfen hast, legst du deine Chips ab und gibst die Würfel nach rechts weiter.  
[Repository](https://github.com/titepasile/ObjektOrientiertesProgrammieren)

## Quellen

* <https://programmieren-starten.de/blog/objektorientierte-programmierung-in-csharp/>

## Anforderungen Elisa Petit-Sirigu

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Muss / Kann?** | **Funktional? Qualität? Rand?** | **Beschreibung** |
|  | Muss | Funktional | Das Programm muss laufen. |
|  | Muss | Funktional | Im Programm soll die Spiellogik implementiert werden. (Erklärung steht unter «Ihr Projekt bei Elisa Petit-Sirigu») |
|  | Muss | Funktional | Das Programm soll den Benutzer fragen, wie der erste Spieler heisst. |
|  | Muss | Funktional | Der Benutzer soll Eingaben tätigen können. |
|  | Muss | Funktional | Nach der Eingabe des Benutzers, soll das Programm fragen, ob der Benutzer erneut einen Spieler hinzufügen möchte. |
|  | Muss | Funktional | Der Benutzer soll mit 1 oder 2 als ja und nein auf die Fragen antworten können. |
|  | Muss | Funktional | Das Programm soll den Benutzer so lange Fragen, ob er einen Spieler hinzufügen will, bis er nein klickt. |
|  | Muss | Funktional | Nach der Eingabe der Spieler soll das Spiel starten. |
|  | Muss | Funktional | Der Spielverlauf soll automatisch abgespielt werden. |
|  | Muss | Funktional | Am Ende muss der Gewinner angezeigt werden. |
|  | Muss | Funktional | Nach dem Spiel soll gefragt werden, ob der Benutzer erneut spielen möchte. |
|  | Muss | Funktional | Der Benutzer soll mit yes oder no das Spiel neustarten oder beenden können. |
|  | Muss | Funktional | Das Programm muss korrekt objektorientiert programmiert werde. |

## Anforderungen Eslem Akdemir

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Muss / Kann?** | **Funktional? Qualität? Rand?** | **Beschreibung** |
|  | Muss | Funktional | Zusammenfassung über das Modul 320 schreiben |
|  | Muss | Funktional | In dieser Zusammenfassung werden Objekte beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung werden Klassen beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung werden Methoden beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung werden Attribute beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung wird UML beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung werden Instanzvariablen beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung wird die Kapselung beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung werden Konstruktoren beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung wird die Vererbung beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung werden Collections beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung werden Assoziationen beschrieben |
|  | Muss | Funktional | In dieser Zusammenfassung wird der Polymorphus beschrieben |

## Testfälle Elisa Petit-Sirigu

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Voraussetzung** | **Eingabe** | **Erwartete Ausgabe** |
|  | Programm offen | Programm starten | Programm läuft ohne Fehler. |
|  | Spielanleitung gelesen | Programm starten | Programm benutzt die Korrekte Spiellogik |
|  | Programm offen | Programm starten | Konsole fragt den Benutzer wie der erste Spieler heisst. |
|  | Erste Frage in der Konsole wird angezeigt | Der Name:  Emilia  *enter* | Eingabe erfolgreich, das Spiel fährt fort. |
|  | Einen Spielername hinzugefügt | *enter* | Konsole fragt den Benutzer, ob er noch einen Spieler hinzufügen will. |
|  | Programm ist gestartet und man hat bereits einen Spieler eingegeben.  Die Konsole fragt, ob man noch einen Spieler hinzufügen möchte. | 1 | Konsole fragt, wie der Spieler heisst. |
| 1.6.1 | Programm ist gestartet und man hat bereits einen Spieler eingegeben.  Die Konsole fragt, ob man noch einen Spieler hinzufügen möchte. | 2 | Das Spiel startet automatisch |
|  | Benutzer hat bei der Frage bei 1.6 1 geklickt und gibt den Namen ein. | Jonas | Die Konsole fragt, ob der Benutzer einen weiteren Spieler hinzufügen möchte. |
|  | Alle Spieler wurden angegeben | *enter* | Spiel startet automatisch. |
|  | Spiel hat automatisch gestartet | Keine Eingabe | Das Spiel läuft von allein und macht alle Spielzüge. |
|  | Spiel läuft automatisch | Keine Eingabe | Gewinner wird automatisch angezeigt. |
|  | Gewinner wird angezeigt | Keine Eingabe | Konsole fragt, ob man noch eine Runde spielen möchte. |
|  | Konsole fragt, ob man noch eine Runde spielen möchte. | yes | Spiel startet von neu |
| 1.12.1 | Konsole fragt, ob man noch eine Runde spielen möchte. | no | Programm schliesst sich |

## Testfälle Eslem Akdemir

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Voraussetzung** | **Eingabe** | **Erwartete Ausgabe** |
| 1.1 | Zusammenfassung ist geöffnet | - | - |
| 2.1 | Zusammenfassung ist geöffnet | Kapitel «Objekte» öffnen | - |
| 3.1 | Zusammenfassung ist geöffnet | Kapitel «Klassen» öffnen | - |
| 4.1 | Zusammenfassung ist geöffnet | Kapitel «Methoden» öffnen | - |
| 5.1 | Zusammenfassung ist geöffnet | Kapitel «Attribute» öffnen | - |
| 6.1 | Zusammenfassung ist geöffnet | Kapitel «Klassen, Attribute, Methoden und Objekte im UML» und «Attribute UML» und «Konstruktoren UML» und «UML Sequenzdiagramme» öffnen | - |
| 7.1 | Zusammenfassung ist geöffnet | Kapitel «Instanzvariablen» öffnen | - |
| 8.1 | Zusammenfassung ist geöffnet | Kapitel «Kapselung» öffnen | - |
| 9.1 | Zusammenfassung ist geöffnet | Kapitel «Objekt Instanzierung und Konstruktoren» öffnen | - |
| 10.1 | Zusammenfassung ist geöffnet | Kapitel «Vererbung» öffnen | - |
| 11.1 | Zusammenfassung ist geöffnet | Kapitel «Collections» öffnen | - |
| 12.1 | Zusammenfassung ist geöffnet | Kapitel «Assoziatonen» öffnen | - |
| 13.1 | Zusammenfassung ist geöffnet | Kapitel «Polymorphus» öffnen | - |

# Planen

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Frist** | **Beschreibung** | **Zeit (geplant)** |
|  | 11.01 | Projektwahl treffen und Dokumentation bis und mit entscheiden fertigstellen. | 2 \* 5 |
|  | 18.01 | Geeigneten Kurs für das Lernen der objektorientierten Programmierung finden. | 2 \* 2 |
|  | **24.01** | Projektbericht schrieben. | - |
|  | 25.01 | Sich mit den Grundlagen der objektorientierten Programmierung befassen. | 2 \* 5 |
|  | **25.01** | Projektbericht schrieben. | - |
|  | 01.02 | Kleines Programm in C# und objektorientierten programmieren. | 2 \* 5 |
|  | 01.02 | Dokumentation Realisieren und Testprotokoll ausfüllen. | 2 \* 1 |
|  | **21.02** | Projektbericht schrieben. **Frist 18.00** | - |
|  | 22.02 | Sich mit fortgeschritteneren Aufgaben beschäftigen. | 2 \* 6 |
|  | **22.02** | Projektbericht schrieben. | - |
|  | 29.02 | Fortgeschrittenes Programm schreiben. | 2 \* 5 |
|  | 29.02 | Dokumentation Realisieren und Testprotokoll ausfüllen. | 2 \* 1 |
|  | **06.03** | Projektbericht schrieben. **Frist 18.00** | - |
|  | 07.03 | Dokumentation Kontrollieren und Auswerten ausfüllen. | 2 \* 1 |
|  | 07.03 | Portfolio schreiben. | 2 \* 4 |

*35 Lektionen pro Person.   
Der Projektantrag muss nicht in den Lernatelierlektionen geschrieben werden, weshalb wir ihn nicht in die Planung mit einbezogen haben.*

# Entscheiden

Elisa Petit-Sirigu

* Währen dem Ablauf des Projekts, habe ich mich dazu entschieden nicht zwei, sondern nur ein OOP-Programm zu programmieren, weil ich keine Ideen für ein kleines Projekt hatte. Deshalb habe ich, als mein «kleines Projekt» den Grundaufbau gemacht und als grosses Projekt mein Gerüst ergänzt.

# Realisieren

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Frist** | **Beschreibung** | **Zeit (geplant)** | **Zeit (effektiv)** |
|  | 11.01 | Projektwahl treffen und Dokumentation bis und mit entscheiden fertigstellen. | 2 \* 5 | 2 \* 5 |
|  | 18.01 | Geeigneten Kurs für das Lernen der objektorientierten Programmierung finden. | 2 \* 2 | 2 \* 2 |
|  | **24.01** | Projektbericht schrieben. | - | - |
|  | 25.01 | Sich mit den Grundlagen der objektorientierten Programmierung befassen. | 2 \* 5 | 2 \* 5 |
|  | **25.01** | Projektbericht schrieben. | - | - |
|  | 01.02 | Kleines Programm in C# und objektorientierten programmieren. | 2 \* 5 | 2 \* 5 |
|  | 01.02 | Dokumentation Realisieren und Testprotokoll ausfüllen. | 2 \* 1 | 2 \* 1 |
|  | **21.02** | Projektbericht schrieben. **Frist 18.00** | - | - |
|  | 22.02 | Sich mit fortgeschritteneren Aufgaben beschäftigen. | 2 \* 6 | 2 \* 6 |
|  | **22.02** | Projektbericht schrieben. | - | - |
|  | 29.02 | Fortgeschrittenes Programm schreiben. | 2 \* 5 | 2 \* 5 |
|  | 29.02 | Dokumentation Realisieren und Testprotokoll ausfüllen. | 2 \* 1 | 2 \* 1 |
|  | **06.03** | Projektbericht schrieben. **Frist 18.00** | - | - |
|  | 07.03 | Dokumentation Kontrollieren und Auswerten ausfüllen. | 2 \* 1 | 2 \* 1 |
|  | 07.03 | Portfolio schreiben. | 2 \* 4 | 2 \* 4 |

# Kontrollieren

## Testprotokoll Elisa Petit-Sirigu

**Testumgebung: Visual Studio Code  
Computer: Windows 11**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nummer** | **Test Nr.** | **Datum** | **Resultat** | **Durchgeführt** |
|  | 1.1 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.2 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.3 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.4 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.5 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.6 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.6.1 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.7 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.8 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.9 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.10 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.11 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.12 | 07.03.2024 | **OK** | Elisa Petit Sirigu |
|  | 1.12.1 | 07.03.2024 | **OK** | Elisa Petit Sirigu |

## Testprotokoll Eslem Akdemir

**Testumgebung: Visual Studio Code  
Computer: Windows 10**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nummer** | **Test Nr.** | **Datum** | **Resultat** | **Durchgeführt** |
| 1.1.1 | 1.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 2.1.1 | 2.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 3.1.1 | 3.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 4.1.1 | 4.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 5.1.1 | 5.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 6.1.1 | 6.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 7.1.1 | 7.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 8.1.1 | 8.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 9.1.1 | 9.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 10.1.1 | 10.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 11.1.1 | 11.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 12.1.1 | 12.1 | 07.03.2024 | **OK** | Eslem Akdemir |
| 13.1.1 | 13.1 | 07.03.2024 | **OK** | Eslem Akdemir |

## Auswerten

Elisa Petit-Sirigu:

## Wie ist das Projekt gelaufen?

## Dieses Projekt war sehr speziell aufgebaut. Weil ich mit meinem Teammitglied nur sehr wenig Kontakt während dem Projekt hatte, fühlte es sich meisten wie ein Einzelprojekt an.

## Ich kam aber gut voran. Der Plan war, ein Kleineres Programm zu machen, bei dem man die Basics wieder repetieren kann und danach ein grösseres, um sich in das OOP zu vertiefen. Allerdings ist mir nichts Gutes für ein kleines Projekt eingefallen. Deshalb habe ich, als mein «kleines Projekt» den Grundaufbau gemacht und als grosses Projekt mein Gerüst ergänzt.

## Diese Aufteilung hat für mich gut funktioniert. In diesem Projekt ist mir die Planung sehr gut gelungen. Ich musste nichts verschieben, war immer im Zeitplan und hatte am Ende sogar noch Zeit übrig. Das ist das erste Projekt, bei dem alles so gelaufen ist, wie ich es geplant habe.

Eslem Akdemir:

Ich konnte alle Anforderungen erreichen und somit ist das Projekt vollständig.