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 |
|  |  |  |  |
| 10.05.2024 | 1.1.0 | Informieren, Planen, Entscheiden, Realisieren, Kontrollieren, Auswerten | Elisa Petit Sirigu, Eslem Akdemir |

# 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 unterschiedlichen Projekten arbeiten und beide auf GitHub veröffentlichen.

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/>
* [LeftCenterRight](https://github.com/titepasile/LeftCenterRight)

## 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. |

## 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 |

# 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.
* Wehrend dem Verlauf des Projektes traten Komplikationen auf. Weil mein Code noch nicht vollständig ist und ich nicht weiss, was ich machen muss, und was fehlt, muss ich das Projekt unabgeschlossen abgeben.
* Ich habe ein par Codestellen vom alten Projekt kopiert, damit ich so viel wie möglich selbst Programmieren kann. Denn ich war von ein par Komponenten, die ich nicht selbst programmieren konnte, abhängig und konnte nicht weiter machen, ohne diese einzusetzen. Ich habe das im Code markiert.
* Ich habe den Teil von Frau Akdemir aus dieser Datei gelöscht, weil ich mein Projekt früher abgeben musste und weil Frau Akdemir später daran gearbeitet hat als ich. Ich konnte nicht auf sie warten und habe dieses Dokument deshalb ohne ihren Teil abgegeben.

# 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 | **NO** | Elisa Petit Sirigu |
|  | 1.2 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.3 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.4 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.5 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.6 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.6.1 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.7 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.8 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.9 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.10 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.11 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.12 | 07.03.2024 | **NO** | Elisa Petit Sirigu |
|  | 1.12.1 | 07.03.2024 | **NO** | Elisa Petit Sirigu |

Fazit: Es gibt Funktionen von den Testfällen, die ich implementiert habe und ich mir sicher bin, dass sie funktionieren, allerdings kann man das Spiel nicht spielen und deshalb sind alle Testfälle auf nicht erfüllt.

## 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 jedoch hatte ich Probleme mit der Umsetzung. Ich konnte mein Projekt nicht fertigstellen, weil ich beim Programmieren nicht mehr weiterwusste und ich einen Fehler habe, den ich nicht wegbekomme.

Wo hatte ich Probleme?   
Ich konnte viele Lücken nicht füllen. Zum Beispiel hat ich ein paar wichtige Komponenten vergessen, wie zum Beispiel den Becher oder das Programmieren vom Weitergeben der Chips nach links und rechts. Ich war etwas verwirrt wehrend dem Programmieren, weil ich öfters den roten Faden verloren habe. Auch am Ende konnte ich den genauen Ablauf des Programms nicht programmieren, weil ich zum einen nicht alle Teile hatte, um das zu machen und weil ich zum anderen nicht wusste, wie ich dabei voran gehen sollte.