

Report Textbasiertes Spiel

In diesem textbasierten Spiel schlüpft der User in die Rolle eines CLT-Tutors an einer Monsterschule. Um am Ende die bestmögliche Evaluation zu erhalten, muss der Tutor seine Aufgaben gründlich erledigen. Hierbei kann der User Energie sammeln, aber auch durch verschiedene Herausforderungen auch verlieren. Wenn der User zu wenig Energie besitzt, verliert er das Spiel.

Das Spiel ist in Python geschrieben worden und gliedert sich in mehrere Dateien auf:

Mit als erste Datei habe ich die Übergangsdatei geschrieben, um ein grobes Konstrukt des Spieles zu konstruieren, welches nach und nach detaillierter wurde.

Das Spiel folgt einem festgelegten Ablauf, der durch verschiedene Zustände repräsentiert wird. Der Spieler interagiert mit dem Spiel, indem er Entscheidungen trifft und Aktionen ausführt. Die Zustände umfassen "Start", "Bewertung 1", "Bewertung 2", "Freizeit", "Snack", "Feedback 1", "Feedback 2", "Evaluation" und "Niederlage". Der Übergang zwischen den Zuständen erfolgt basierend auf den Spielerentscheidungen und dem Spielverlauf.

Sobald die Spielerentscheidung erfasst worden ist, wird je nach Zustand verschiedene Handlungen und Textausgaben ausgeführt. Gerade am Anfang habe ich den Fehler begangen bei der Implementierung des Codes vor jedem Zustand If-Schleifen anstelle von Elif-Schleifen zu verwenden. Dadurch sind einige Fehler aufgetreten und der Code hat nicht so funktioniert wie ich mir das gewünscht hätte oder ist abgebrochen. Durch die Elif-Schleife konnte ich diesen Fehler nun lösen.

Ein weiteres Problem, welches aufgetreten ist, war, dass man bei Bedarf zu vorherigen Zuständen zurückkehren kann, ohne dass der Code wiederholt wird oder man von vorne anfangen muss. Das ist zum Beispiel beim Wechsel zwischen dem Zustand „Freizeit“ zurück zum Zustand „Bewertung 1“ gewesen. Lösen konnte ich dieses Problem, indem ich aktuelle Zustände in einer Liste gespeichert habe, um so den richtigen Zustand zu reaktivieren und fortzusetzen.

Nachdem ich ein grobes Gerüst für das Spiel hatte, habe ich mich an die einfacheren Spielelemente gewagt und angefangen einen Spieler zu initialisieren.

Dafür habe ich mich zunächst mit objektorientiertem Programmieren beschäftigt. Das hat mir das Gefühl gegeben, die Elemente besser miteinander verknüpfen zu können und so einfacher mit den Elementen arbeiten zu können.

Für den Spieler habe ich eine Klasse erstellt, mit wichtigen Funktionen, auf die auch innerhalb der Übergangsdatei zugegriffen werden kann. Eine wichtige Funktion ist zum Beispiel die Funktion „Bewertung“, bei der je nach Wahrscheinlichkeit eine andere Funktion aktiviert wird. In diesem Fall habe ich etwas über die Wahrscheinlichkeit grübeln müssen, mich am Ende aber dafür entschieden, eine Zahl zufällig zu generieren und je nach Zahl eine Funktion zu aktivieren.

Am schwierigsten innerhalb dieser Datei war die Überlegung, ob ich aus `Inspect_Report` eine Funktion oder einen Zustand machen soll. Ich habe beides ausprobiert und am Ende festgestellt, dass eine Funktion flüssiger innerhalb der Übergänge arbeitet und zudem weniger Code braucht. Dadurch konnte ich mir auch sparen nochmals in den vorherigen Zustand zurückkehren zu müssen, nur um eine weitere Abfrage zu machen und den Code noch einmal durchführen zu müssen.

Danach habe ich mich an der Einmal-Funktion versucht.

Obwohl dieser Code sehr kurz ist, gab es doch Probleme beim Implementieren. Hierbei war die größte Herausforderung einen anderen Code nur ein einziges Mal ausführen zu lassen und die Einmal-Funktion mehrere Male zu benutzen, ohne dass ich diese doppelt aufschreiben muss. Dieses Problem habe ich durch die Verwendung einer globalen Variable lösen könne, welche ich innerhalb der Zustände so anpassen konnte wie es notwendig war, damit der Code nicht zu oft ausgeführt wird.

Am meisten Kopfschmerzen hat mir jedoch das Feedback 1 gemacht.

Hierbei war die Aufgabe zwar von Anfang an klar und die Implementierung des Codes für die Kategorie hat auch gut funktioniert, jedoch hatte ich Probleme, dass der Countdown innerhalb der Durchführung parallel zur Eingabe verläuft und nicht nacheinander.

Am Ende konnte ich das durch genügend Internetrecherche und dem Sprechen mit anderen Leuten lösen.

Ein weiteres Problem bei der Durchführung war auf die Anzahl der Versuche zu achten und gegebenenfalls das Thema zu ändern, um die Feedbackrunde zu bestehen.

Lösen konnte ich dieses Problem durch eine While-Schleife und dem Initialisieren von drei verschiedenen Kategorien, welche innerhalb der While-Schleife bei dem Richtigeraten eines

Themas geändert werden konnte.

Am meisten Spaß hat jedoch das Implementieren des Feedbacks 2 gemacht, auch wenn ich dieses am schwierigsten in meine Storyline integrieren konnte.

Das einzig größere Problem, welches hier aufgetreten ist, war wo die eingegebenen Wörter gesplittet werden und somit einen gültigen Codesatz ergeben oder nicht. Herausfinden konnte ich das durch Ausprobieren, indem ich Print-Statements verwendet habe und so den Output der gesplitteten Wörter überprüfen konnte. Dadurch konnte ich eingrenzen, was gespalten wird und was nicht und somit meinen Code anpassen.

Nachdem mein Code „fertig“ war, hat sich ein Freund von mir, der keine Ahnung, von dem Projekt hat, dazu bereit erklärt das Spiel für mich zu testen. Hierbei konnte ich die Spielweise noch einmal aus einer anderen Perspektive nachvollziehen und kleinere Fehlerquellen wie Schreibfehler oder auch größere als bei den Übergängen mit der Elif-Schleife herausfiltern und korrigieren. Dadurch ist mir auch aufgefallen, dass einige Funktionen nicht funktionierten, da ich vergessen hatte Klammern hinter diese zu schreiben. Nachdem ich größere Fehler korrigiert hatte, haben wir das Spiel noch mehrere Male getestet, um so Fehler zu minimieren und ich konnte so ein funktionierendes Spiel programmieren.

Zusammenfassend hatte ich doch viel Spaß an dem Spiel zu arbeiten und ich konnte so einiges mitnehmen. Gerade die Verwendung von Zuständen und dem Implementieren einer Einmal-Funktion war entscheidend für den Erfolg des Projektes. Durch die Lösung der identifizierten Probleme und die Möglichkeit einen „Beta-Tester“ zu haben konnte das Spiel reibungslos ablaufen.

In Zukunft könnten weitere Herausforderungen zum Spiel hinzugefügt werden. Ändern würde ich wahrscheinlich auch die Feedbackrunden, da mein Freund gerade bei dem Erraten der Themen in Feedbackrunde 1 Schwierigkeiten hatte das Thema zu identifizieren. Hier müssten entweder die Kriterien geändert, angepasst oder das Thema eingrenzt werden.

