

Textadventure selbstgemacht mit Quest 5 und Squiffy

Description

Ein Textadventure ist so ziemlich die Urform des Computerspiels. Bereits um 1976 erschuf William Crowther ein Textabenteuer namens **Adventure**. Es war mit Fortran IV programmiert und lief auf damaligen Großrechnern… ein Genre war geboren!

Heute blicken wir auf eine Reihe von Textadventures zurück. Vielen dieser Spiele wurden Grafiken hinzugefügt, doch gerade die Entwickungsfirma „Infocom" produzierte Games in reiner Textform. Da gab es die **Zork-Serie** (**Zork I** erschien 1980) und Games wie **Planetfall** (1983) mit dem Nachfolger **Stationfall** (1987) . Wer sich für die Geschichte der Adventures interessiert, dem sei die Artikelreihe von Sven Gramatke wärmstens empfohlen.

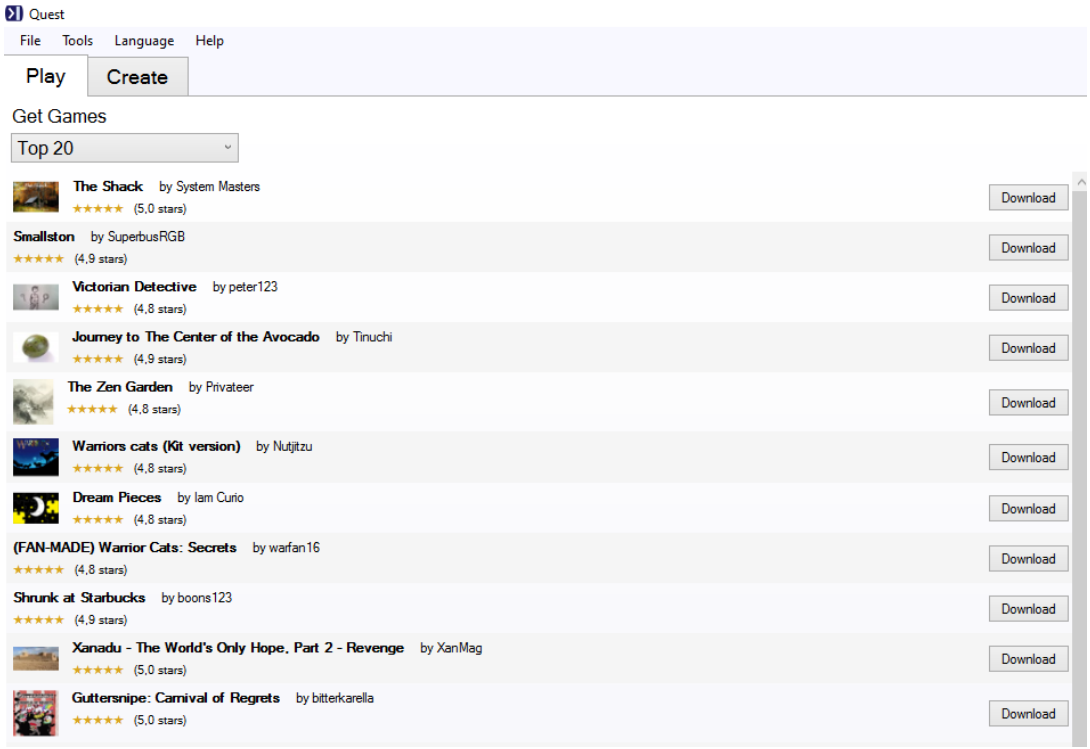
Vom Ende des Booms

Doch der Boom war nur von begrenzter Dauer. Grafische Anwendungen und Spiele traten ihren Siegeszug an. Neuerscheinungen reiner Textadventures wurden immer seltener. Ab den frühen 90ern hatten die meisten Textadventures zumindest Standgrafiken dabei. Die Möglichkeiten der animierten grafischen Darstellung überrollten jedoch das Genre und die Ansprüche der Spielerschaft änderten sich ebenfalls. **Thaumistry** erschien noch 2017, finanziert durch Crowdfunding. Dies war eines der letzten größeren Text/Grafik-Adventures und eine Hommage an vergangene Zeiten.

Und jetzt? Sind Textadventures ausgestorben wie damals schon die Dinosaurier? Nicht ganz. Es gibt noch kleine Text-Abenteuer-Inseln. Eine dieser Inseln findet man auf textadventures.co.uk mit einer kleinen aber aktiven Community. Hier werden mit den Tools **Quest** und **Squiffy** spannende textbasierte Abenteuerspiele erstellt, hochgeladen und geteilt. Die Tools sind in einer online Version und als Client verfügbar – da bei mir die Online-Version hing, probierte ich zuerst den Client für Windows von **Quest 5**. Der Editor für **Quest** zum Download gibt es nur für Windows. Den **Squiffy-**Editor gibt es für Windows, OS X und Linux und erreicht damit auch User ohne Windows-System.

Textadventure oder Gamebook?

Generell unterscheidet man bei Textadventures zwischen 2 Arten: das klassische Textadventure mit Texteingabe und das Gamebook nach dem Vorbild der Abenteuer-Bücher, die in den 80er Jahren ihre Ära hatten. Bei diesen Spielbüchern werden dem Spieler zwei oder mehr Optionen angezeigt, zwischen denen er sich entscheiden kann. Mit **Quest 5** erhält man einen Editor für beide Modi, wohingegen Squiffy nur das Gamebook abdeckt.

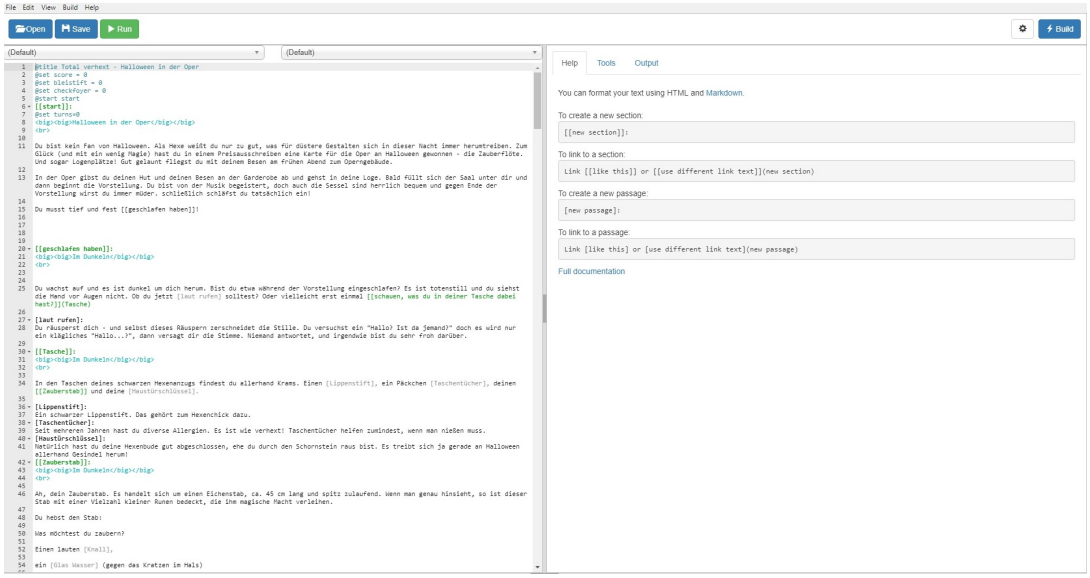


Mit dem **Quest-Client** können diverse mit Quest erzeugte Spiele heruntergeladen werden. Leider kann man diese nicht nach Landessprache filtern.

Was mich bei englischen Textadventures immer abgeschreckt hatte, war die Sprachbarriere. Zwar hatte ich auch Englisch in der Schule, aber um die Feinheiten zu verstehen und die textbasierten Rätsel eines Adventures zu lösen, fehlte es mir an Vokabular. Quest 5 hingegen bietet einen komplett ins Deutsche übersetzten Editor an. Sowohl die Oberflächen als auch die Ingame-Meldungen sind deutschsprachig und ermöglichen somit die Erstellung deutschsprachiger Textadventure.

Gamebook-Spiele

Squiffy ist ein Produkt der gleichen Entwickler und ebenfalls auf textadventure.co.uk zum Download erhältlich. Es ist deutlich simpler gestrickt – und exportiert auf jedem Webserver laufende Html/CSS/JS dateien. Mit **Squiffy** erstellt man keine Parser-Spiele, sondern eher Multiple-Choice-Games in der Art von Spielbüchern. Dadurch wird ein Zurückgehen in der Spielumgebung oft nicht möglich. Dies führt dazu, dass die Story anders konzipiert werden muss.



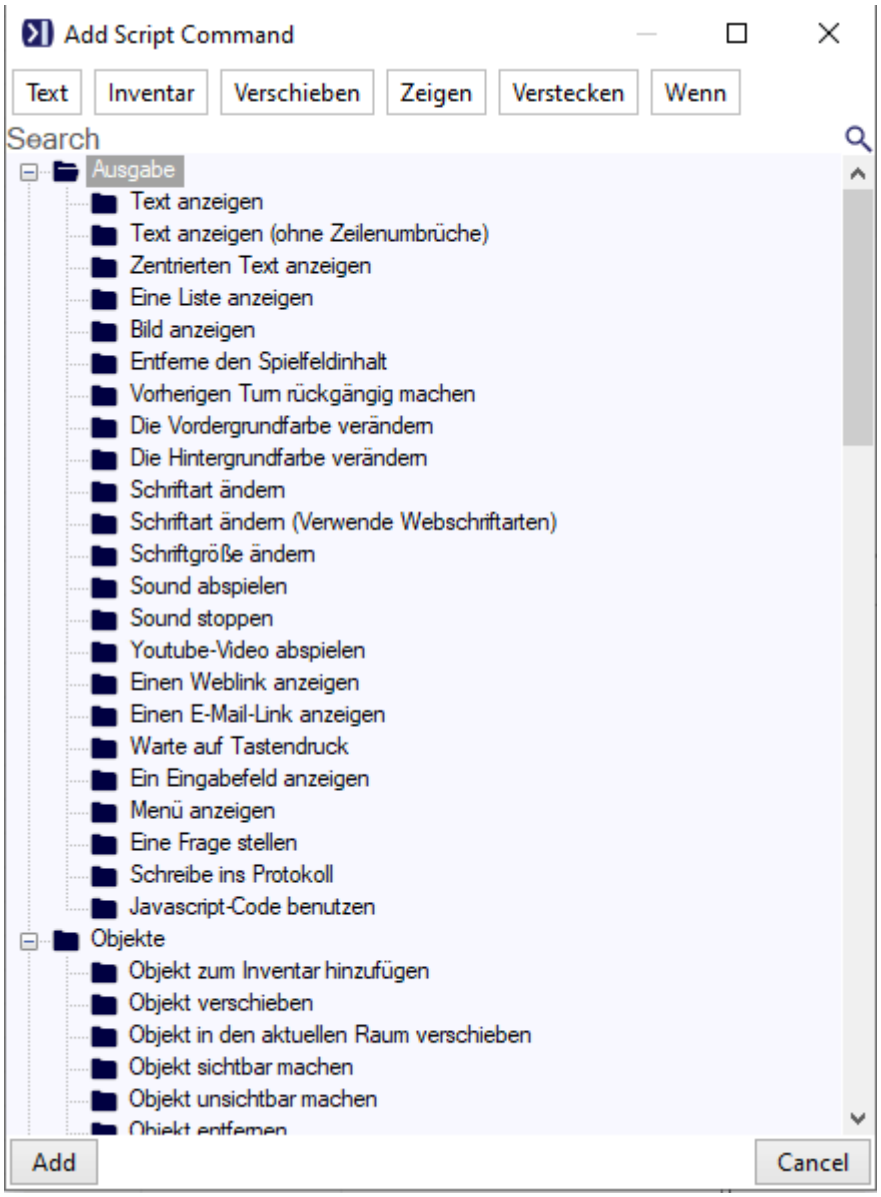
Der **Squiffy**-Editor besteht aus einem großem Eingabefeld und einer Testoberfläche.

Mein größter Fehler am Anfang mit **Squiffy** war, eine **Quest 5**-Geschichte eins-zu-eins auf **Squiffy** übertragen zu wollen. Das führt zu unschönen Effekten. Wenn ich in **Quest 5** zum Beispiel einen Schlüsselgegenstand finden kann, den ich dringend brauche, um weiterzukommen, ist das ok. Ich kann jederzeit zu dem entsprechenden Raum und den Gegenstand finden. Bei **Squiffy** führen solche unverzichtbaren Schlüsselgegenstände eventuell zum vorzeitigen Spielende. Wenn man sie einmal verpasst hat, war es das. Deswegen muss man bei der Spielgestaltung anders vorgehen, wenn man dem Spieler kein hartes Try-and-Error zumuten möchte.

Spiele in **Quest** bieten also mehr Freiheiten und komplexe Puzzles, während bei **Squiffy** es mehr um das Erzählerische geht. Beides hat vor und Nachteile.

Kann man das Scripten?

Quest ist mit C# programmiert, verwendet aber hauptsächlich XML oder ASL, eine spezielle Scriptsprache mit Ähnlichkeiten von C++ und Java – mit Anleihen an Visual BASIC. Wer jetzt einen Schreck bekommen hat und denkt, man müsste erst Informatik studieren, sei beruhigt. **Quest 5** besitzt relativ leicht verständliche Script-Formulare, die es auch mit geringen Grundkenntnissen ermöglichen, das Adventure aufzupeppen. Allerdings ein wenig Programmiererfahrung ist schon sehr hilfreich, wenn man mit dem Scripten so richtig loslegen möchte. Wer lieber manuell scripted statt die Formulare zu benutzen, kann jederzeit auf eine reine Code-Ebene umschalten.



Quest 5 bietet zahlreiche Formulare an, welche das Scripting deutlich vereinfachen.

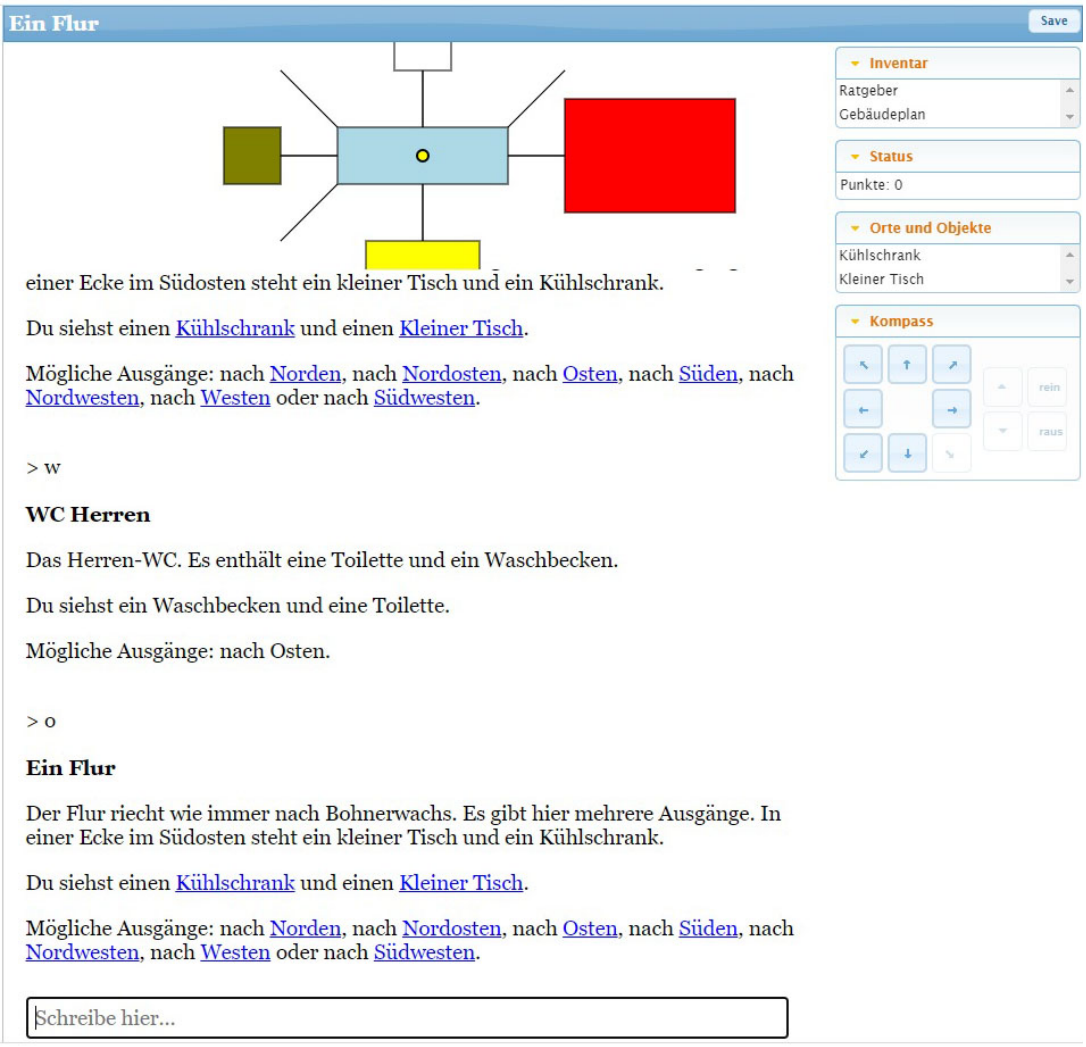
Die diversen Oberflächen bieten umfangreiche Möglichkeiten. Objekte und Verben und deren Wechselwirkungen können definiert werden. Textausgaben oder Aktionen lassen sich über die Script-Formulare bequem umsetzen. Das ganze ist vielleicht nicht kinderleicht, aber für die gebotene Komplexität durchaus aufgeräumt und benutzerfreundlich. Fertige Produkte können auf der Website hochgeladen und von der Community geprüft werden. Nach einer erfolgreichen Beta-Phase erhält das Spiel einen Platz im Download-Pool der abgeschlossenen Entwicklungen. Nun kann das fertige Projekt vom **Quest-Client** per Mausklick weltweit heruntergeladen werden.

Squiffy bietet wenig Scripting

Die Scriptmöglichkeiten von **Squiffy** im Editor sind eher mager. Es lassen sich ein paar Flags setzen oder Texte manipulieren, aber der Ansatz ist ein gänzlich anderer. Spielpunkte oder Turns lassen sich noch einfach verwalten, auch Flags sind schnell gesetzt und unproblematisch. Allerdings diverse

Gesprächsoptionen wie bei **Quest** oder Kombinationsmöglichkeiten von Gegenständen lassen sich nicht so gut umsetzen bzw. sind nahezu unmöglich. Es sei denn – und hier wird es für Javascript-Programmierer interessant – man bindet JS-Code bei **Squiffy** ein. Damit ist dann natürlich sehr viel möglich, wurde allerdings von mir nicht getestet.

Die etwas reduzierten Scriptmöglichkeiten in **Squiffy** sind jedoch nicht wirklich von Nachteil, da der ganze Spielaufbau ein anderer ist. Was vom Editor mitgeliefert wird, reicht für eine Interaktive Geschichte völlig aus. Somit ist der **Squiffy**-Editor im Vergleich zum **Quest 5**-Editor sehr einfach, ja geradezu schlank, gehalten.



Quest5-Abenteuer besitzen auch ein Automapping.

Eigene Erfahrungen

Auf **Quest 5** bin ich durch den 6. Teil von Sven Gramatkes Adventure-Serie hier bei bytegame.de aufmerksam geworden. Ein deutschsprachiger Editor für Textadventures? Das musste ich ausprobieren! Doch Anfangs tat ich mich doch noch etwas schwer mit der Erstellung meines eigenen Textadventures. Meine Story-Ansätze waren zu Beginn einfach nicht Genre-gerecht. Zuerst wollte ich ein RPG programmieren. Mangels einer deutschen Kampferweiterung für **Quest 5** (es gibt leider nur

eine englische) fiel das jedoch schnell ins Wasser.

Bei meinem nächsten Versuch setzte ich eine Scifi-Story zu umfangreich an. Schnell merkte ich, dass mir der rote Faden aus den Händen glitt. Mein dritter Versuch war viel bescheidener. Er bestand nur aus einer Handvoll Räume und wenigen Personen und hat eine sehr eingegrenzte Story. Dennoch schüttelt man auch ein überschaubares Textadventure nicht mal eben aus dem Ärmel. Dafür steckt da einfach zu viel Arbeit drin.



„Völlig verhext – Halloween in der Oper" – mein zweites Projekt mit Quest 5

Ganz ohne Lags

Zu **Squiffy** bin ich gekommen, weil an manchen Tagen ein Spielen von **Quest 5**-Spielen online nahezu unmöglich war. Haarsträubende Lags von 1-2 Minuten oder komplette Freezes, zudem ein Sessionverlust bei einer Spielpause. Das ganze wollte ich meinen Testern nicht länger zumuten. Da der **Squiffy**-Build rein auf Html/CSS/JS basiert, ist die Serverlast eines solchen Games ungleich geringer. Im Prinzip lädt der Browser einmalig ein paar Dateien – und der Rest funktioniert auf Clientseite. Das macht ein Online-Spielen flüssiger und ich kann es auf jedem Webspace selber hosten. Nachdem ich bei meiner Portierung der „Völlig verhext"-Story auf **Squiffy** zuerst gescheitert

war, lernte ich, den Spielverlauf an die Möglichkeiten der Engine anzupassen. Wenn man **Squiffy**-Spiele machen möchte, muss man eben auch in **Squiffy** denken. Nachdem ich dies Verstanden hatte, gelang es mir auch, mein erstes kleines Spiel in **Squiffy** abzuschließen.



„Völlig verhext" auf Squiffy portiert – der Spielverlauf musste angepasst werden.

Es gibt auch Bugs

Bei meiner Arbeit mit Quest 5 entdeckte ich auch einige Bugs. In der deutschen Übersetzungsdatei befanden sich zum Beispiel noch 2 veraltete Templates für den Parser, welche die Verben „Frage … nach" und „Rede … über" blockierten. Damit waren nahezu alle Gesprächsoptionen mit NPCs in der deutschen Version verbuggt. Die entsprechenden fehlerhaften Zeilen konnten aber mit Hilfe engagierter User aus dem **Quest-Forum** schnell gefunden werden. Ein Auskommentieren der Übeltäter beseitigte das Problem. Ich hoffe natürlich, dass bei der nächsten Version dieser Fehler berücksichtigt wird. Wer **Quest 5** ausprobieren möchte und an dem Bug hängt, kann sich gerne hier in den Kommentaren melden. Ich helfe da gerne.

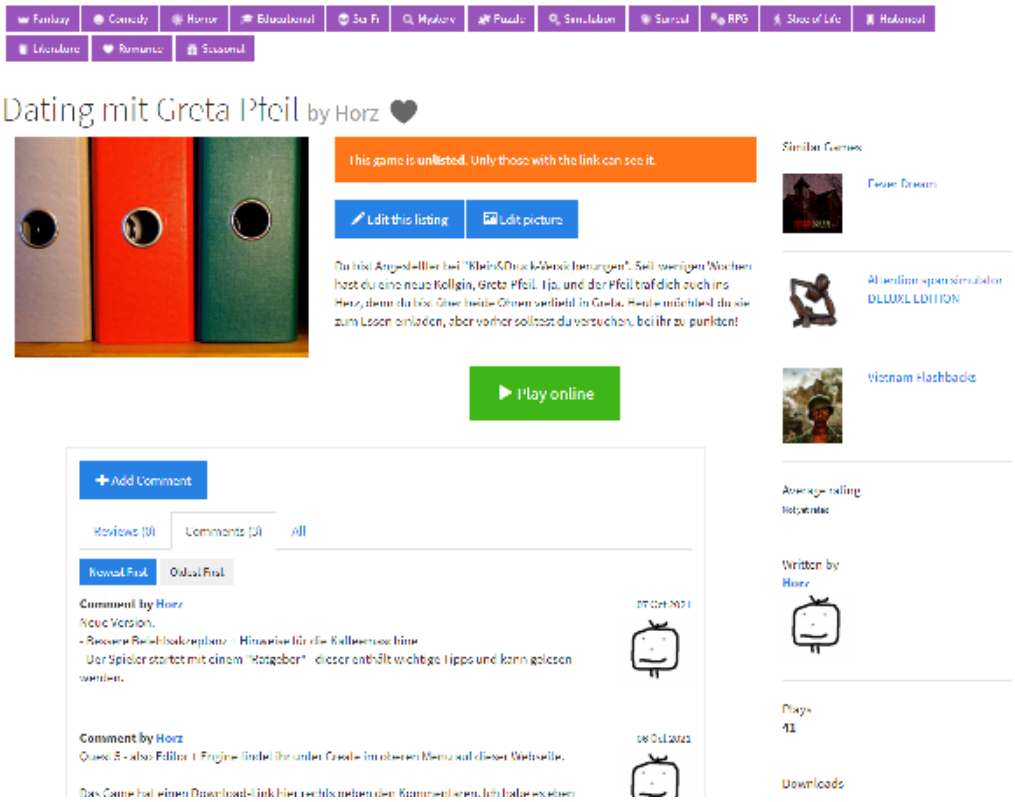
Da der Editor von **Squiffy** viel simpler gehalten ist, kommt es da auch zu weniger Bugs bzw. ich habe bei meiner Arbeit keine festgestellt. Ob der Editor allerdings alles Javascript versteht, was man dort einhacken kann, wage ich nicht zu beurteilen. Die Erstellung und das Editieren, das Probespielen und die Builds liefen jedoch schnell und problemlos.

Online mit Hindernissen

Das fertige Werk kann auch bei **Quest 5** im Editor leicht exportiert werden. Man erhält allerdings eine Game-Datei mit der Endung „.quest", welche alle Ressourcen und Zusatzdateien bündelt. Diese Datei ist nicht ohne einen **Quest 5**-Client lauffähig, allerdings kann diese Game-Datei kann auf textadventure.co.uk hochgeladen werden und steht dort dann für ein Online-Gaming zur Verfügung. Leider ist in den letzten Tagen das Online-Spielen kaum möglich. Es gibt diverse Lags und Freezes,

die dazu führen, dass man immer wieder Neustarten muss. Zum Glück kann das Game-File von dieser Seite auch heruntergeladen werden, so dass man zumindest eine zentrale Downloadseite hier zur Verfügung hat.

Unschöner weise ist der **Quest 5**-Client nur für Windows erhältlich. Das ist sehr schade. Ich hoffe sehr, dass das Online-Spielen der **Quest**-Textadventures bald wieder ohne Freezes möglich sein wird. Nicht jeder Interessierte ist bereit oder in der Lage, sich die Windows-Engine und das Game herunterzuladen, zumal der Download ebenfalls nur mit guten Worten funktionierte (Link kopieren, separat aufrufen).



Jedes Spiel hat auf textadventure.co.uk eine eigene Landing-Page mit Kommentaren, Zugriffswerten und Download-Link.

Die richtigen Worte finden

Insgesamt sind beide Editoren für mich intuitiver und einfacher zu bedienen als zum Beispiel eine **Unity-Engine**. Da ich jedoch wenig Erfahrung mit reinen Textadventures gesammelt habe, fällt mir die Umsetzung einer Idee nicht leicht. Zwar habe ich für 2 Wochen das textbasierte MUD **Morgengrauen** gespielt. Dies war jedoch eher ein Text-MMO als ein reines Adventure. Das die Konzeptionierung eines Textadventures tricky sein kann, vermutet man vielleicht nicht auf anhieb. Natürlich sind ein paar Beschreibungstexte schnell getippt. Aber um daraus eine atmosphärisch dichte Umgebung zu kreieren, in welcher der Spieler gerne abtaucht, bedarf es Erfahrung mit anderen Projekten. Oder man benötigt Übung mit den eigenen.

Das Abenteuer wartet

So bastele ich hin und wieder gerne an meinem bescheidenen Adventure und freue mich über die kleinen Fortschritte. Dabei geht es mir wie vielen Menschen, die sich zum ersten mal mit der Produktion einer bestimmten Sache beschäftigen: Der Respekt vor der Arbeit anderer wächst! Während ich früher oft dachte, dass ein Textadventure relativ simpel von einer Einzelperson mal eben zusammengestrickt wird, sehe ich das heute mit anderen Augen. Aber mit **Quest 5** und **Squiffy** habe ich kostenfreie Tools an der Hand, die zwar Wünsche offen lassen, aber mit denen man Arbeiten kann. Und das eigentliche Abenteuer ist doch immer noch dessen Entwicklung!

Links

Serie: Geschichte der Adventure

Teil 1 – Am Anfang war das Wort
Teil 2 – Der Berg ruft
Teil 3 – Lucas und die SCUMM-Lokomotive
Teil 4 – Fotorealistisch?
Teil 5 – Aufbruch in neue Abenteuer
Teil 6 – Zurück in der Gegenwart

Textadventure

Halloween in der Oper – erstellt mit Squiffy
Zork Anthology bei Steam
Thaumistry bei Steam
MUD Morgengrauen

Quest 5 und Squiffy

textadventures.co.uk

Titelbild

Bild von Kranich17 auf Pixabay

Date Created
15. November 2021
Author
harald