Dokumentation Toni Kostal

|  |  |
| --- | --- |
| Teammitglieder | Toni Kostal  Robin Steiger  Henri Schmekies  Fabian Eichenberger |
| Projektleiter | Henri Schmekies |
| Kunde | M. Schneider |
| Beschreibung Projekt | Ein Game mit Löve2D erstellen und sich damit in die Programmiersprache von Löve2D (Lua) einarbeiten. |
| Datum | 22. April 2015 |
| [🡨 Zurück zur Übersicht](../projektuebersicht.docx) |  |

Inhaltsverzeichnis

[Metainformationen 3](#_Toc415649648)

[Einstieg Lua 4](#_Toc415649649)

[Konfiguration 4](#_Toc415649650)

[CheckCollsion 4](#_Toc415649651)

[Love.load 4](#_Toc415649652)

[Love.update 4](#_Toc415649653)

[Love.draw 5](#_Toc415649654)

[SnakeGame 6](#_Toc415649655)

[SnakeGame erstellen 6](#_Toc415649656)

[Conf.lua 6](#_Toc415649657)

[Game.lua 7](#_Toc415649659)

[Keyboard.lua 8](#_Toc415649660)

[Main.lua 9](#_Toc415649661)

[pellet.lua 10](#_Toc415649662)

[render.lua 10](#_Toc415649664)

[Snake.lua 11](#_Toc415649666)

[tile.lua 12](#_Toc415649668)

[SnakeGame fertiggestellt 12](#_Toc415649670)

[Startmenü 12](#_Toc415649671)

[InGame 13](#_Toc415649672)

[GameLost 13](#_Toc415649674)

Metainformationen

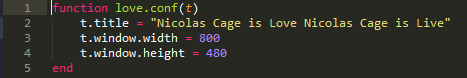
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Datum | Status | Version | Autor | Bemerkung |
| 25.02.2015 | In Bearbeitung | 1.0 | Toni Kostal | Start Projekt |
| 25.03.2015 | In Bearbeitung | 1.0 | Toni Kostal | Einstieg Lua |
| 01.04.2015 | Freigegeben | 1.0 | Toni Kostal | Abgeschlossen und kontrolliert |

|  |  |
| --- | --- |
| Aktueller Status: | Freigegeben |

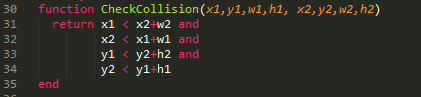
Einstieg Lua

Ich habe noch keine Erfahrung mit Lua und habe mich deshalb zuerst über die wichtigsten Sachen informiert. Für die Grundlagen habe ich hauptsächlich die [Tutorial-Seite](http://nova-fusion.com/2012/08/27/lua-for-programmers-part-1/) benutzt.  
Danach versuchte ich paar Sachen auszuprobieren, wie die Steuerungen, Konfigurationen, Hintergründe, etc. funktionieren. Hierfür benutze ich die [Löve2D-Wiki](https://www.love2d.org/wiki/Main_Page) Seite.

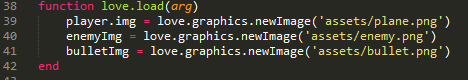
Konfiguration



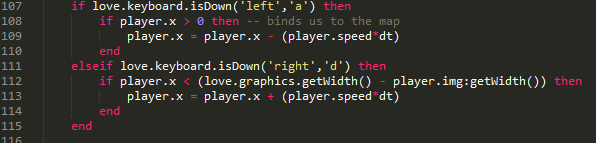
CheckCollsion

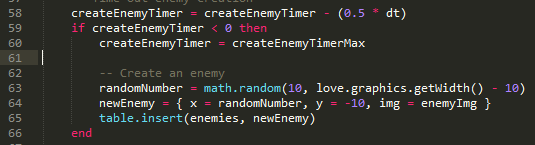
  
der Spieler kann nicht über den Rand hinausgehen.

Love.load

  
lädt die Bilder

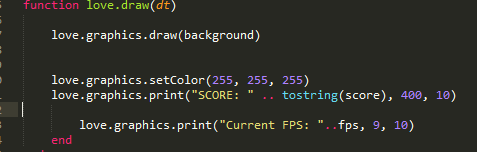
Love.update

  
Spieler kann nur nach rechts oder links gehen

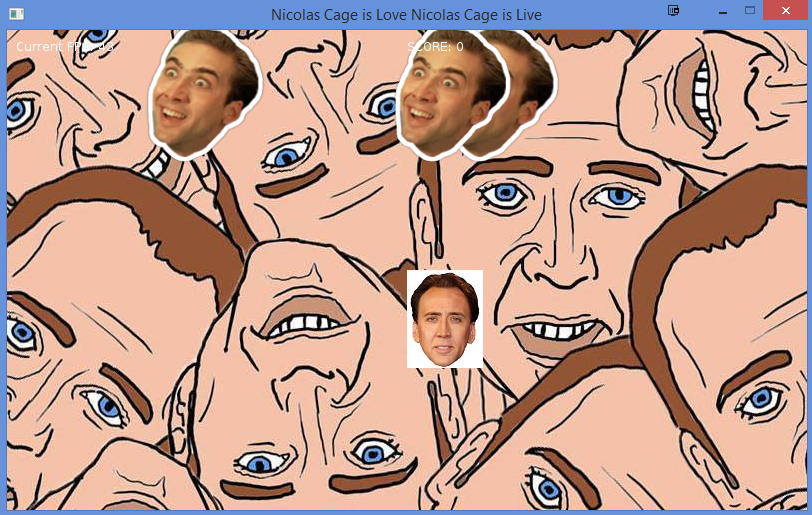


Erstellt einen Gegner

Love.draw

  
erstellt Hintergrund und zeigt Score und FPS an

Hier sieht man, was dabei entstanden ist.



Da ich nun ungefähr wusste, wie lua funktioniert, begann ich mit meinem SnakeGame.

SnakeGame

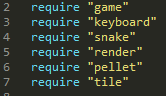
Da jeder der Mitglieder sein eigenes Spiel erstellen sollte, entschied ich mich für den Klassiker „Snake“. Ich will das Spiel in drei Schwierigkeitsgraden einteilen und zwar einfach, mittel und schwer. Dabei sollte je nach Schwierigkeitsstufen das Spiel schneller oder langsamer ablaufen.  
Man muss sogenannte pellets einsammeln, damit das Snake länger wird. Mit jedem pellet, dass man eingesammelt hat, wird unser Score um eins erhöht. Die pellets werden zufällig generiert.  
Nachdem man den Rand oder sich selber berührt hat, erscheint eine Anzeige, welche die Punkte anzeigt und den Schwierigkeitsgrad auf den man gespielt hat. Es wird auch noch gefragt ob man erneut spielen will oder aufhören möchte. Man kann leider nur drei Mal das Spiel erneut spielen, da ich den Bug nicht gefunden habe. Deshalb muss man das Spiel nochmals starten, falls man weiterspielen will.

SnakeGame erstellen

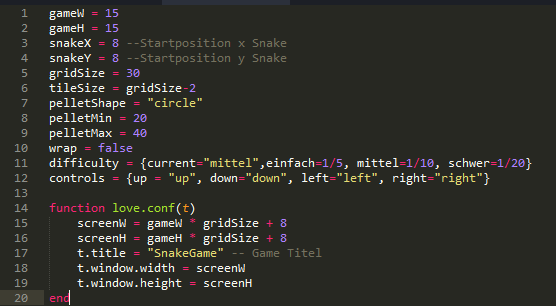
Um die Aufgabe ein bisschen zu erleichtern, habe ich einen [Code](https://github.com/2bt/Beer-Snake/blob/master/l%C3%B6ve/main.lua) eines anderen Snake-Games als Beispiel genommen.

Damit das Ganze ein bisschen übersichtlicher gestaltet ist, habe ich es einfach aufgeteilt.

Main.lua braucht dann folgendes



Conf.lua

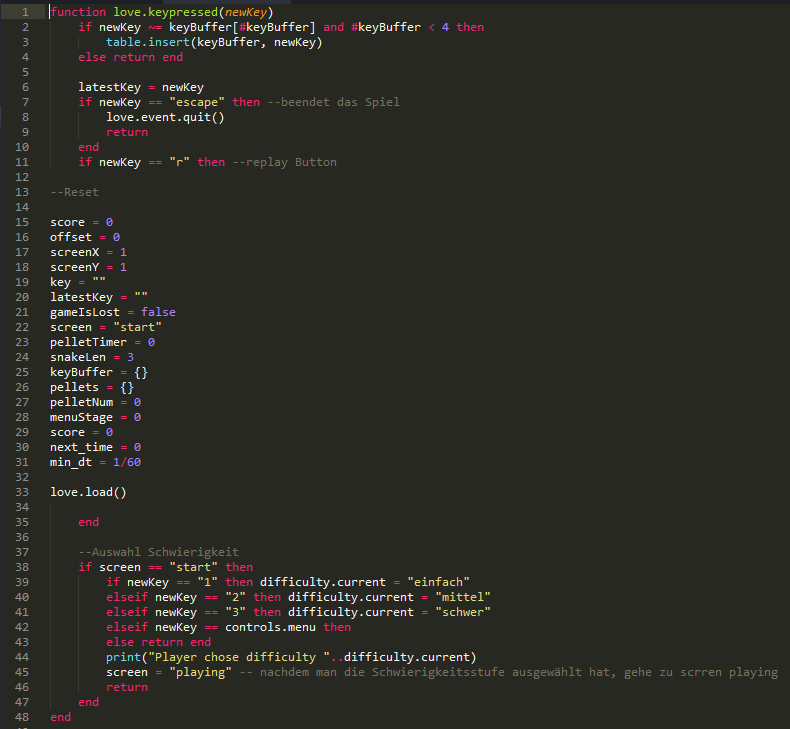


Game.lua



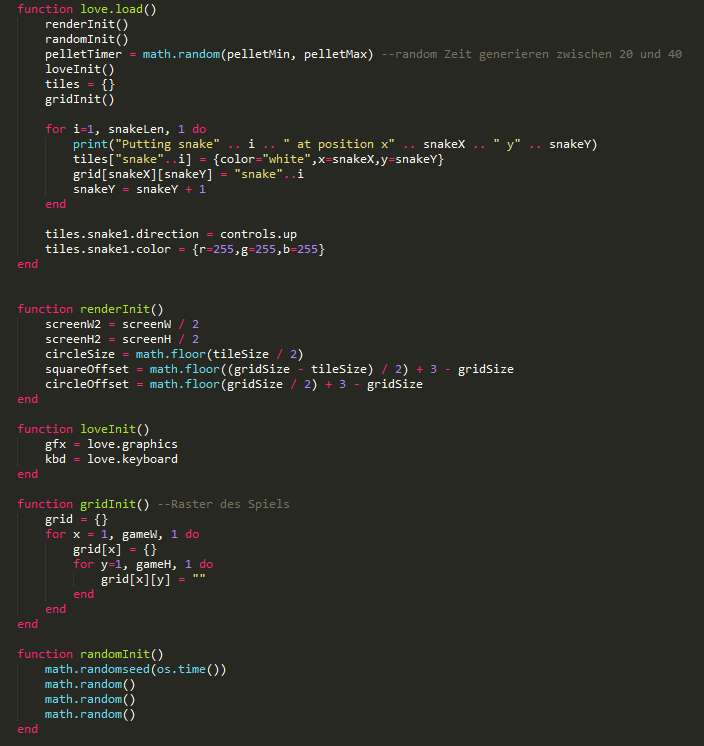
Hier sind die Steuerungen des Spiels enthalten und die Funktion für loseGame

Keyboard.lua



Wenn man „escape“ drückt, wird das Spiel beendet und wenn man „r“ drückt wird das Spiel neugestartet. Hier ist auch festgelegt mit welcher Taste man die Schwierigkeitsstufen auswählen kann.

Main.lua



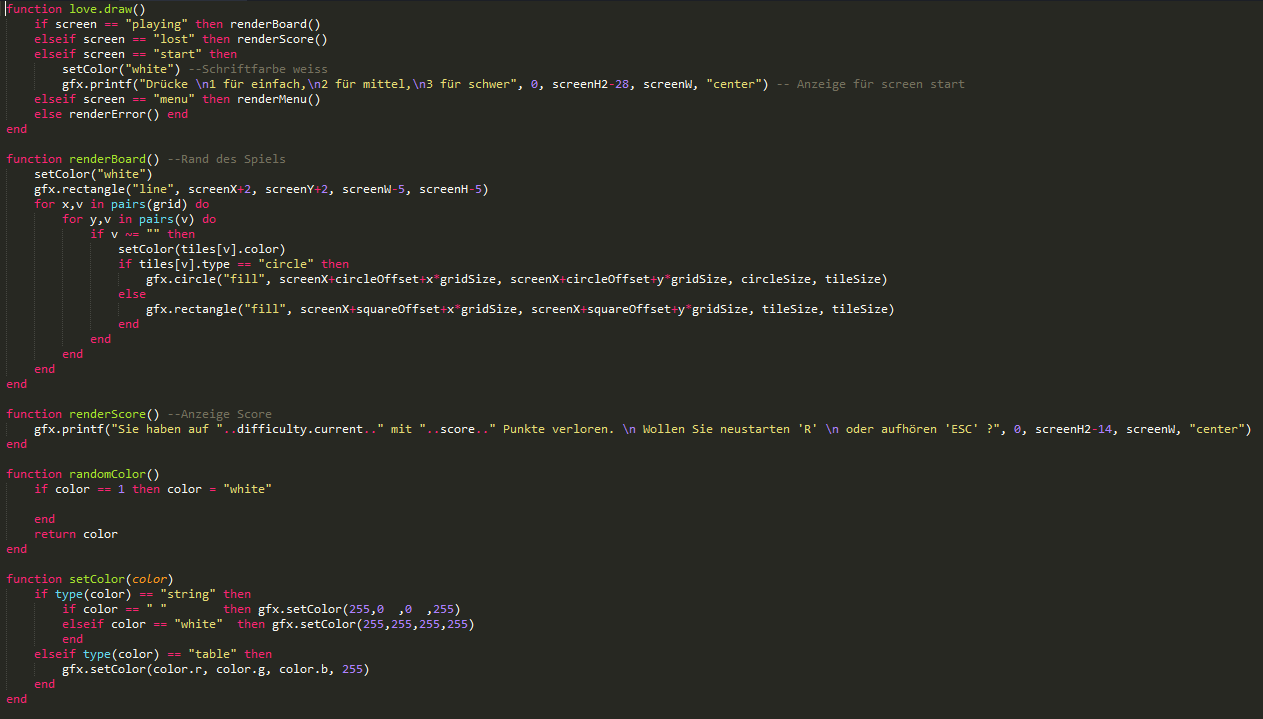
Wird festgelegt, wo der Spieler starten muss.

pellet.lua



pellet wird an einem zufälligen Ort generiert

render.lua



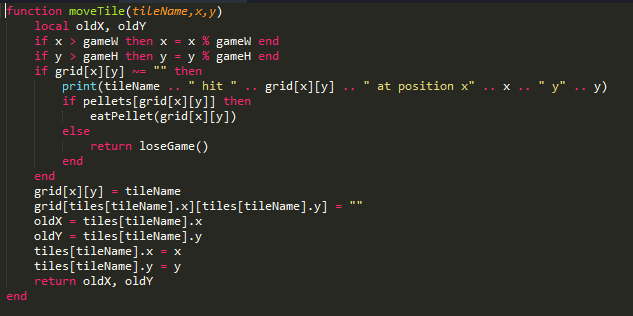
Wird festgelegt in den verschiedenen screens passieren sollte

Snake.lua



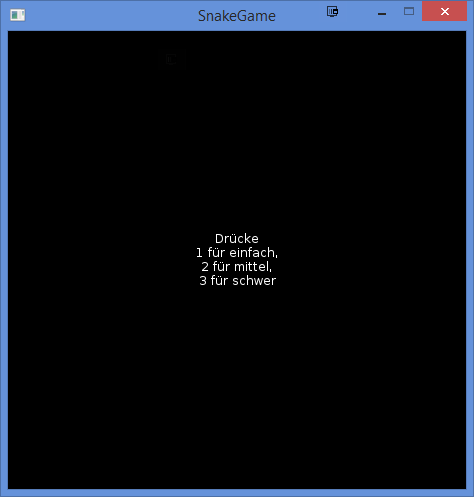
Wenn Snake Rand berührt, wird screen „lose“ angezeigt.  
Wenn Snake pellet berührt, wird die Länge und Score um eins grösser.

tile.lua

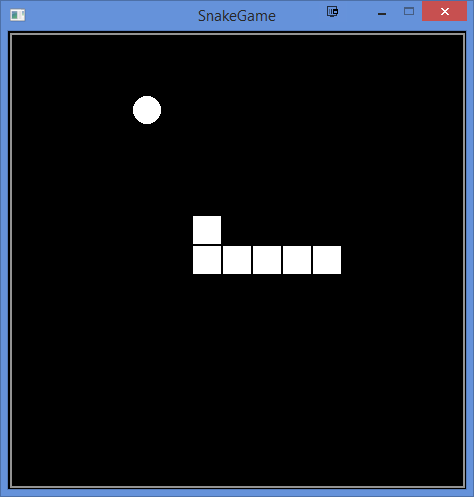


SnakeGame fertiggestellt

Startmenü



InGame



GameLost

