

# Projekt: Navigation durch ein Labyrinth

## 1. Erstellen der Objekte

Um in den Etoys ein Objekt zu erstellen muss es mit **dem Malwerkzeug gemalt** werden. Sollen mehrere, voneinander unabhängige Objekte erstellt werden, müssen diese auch nacheinander einzeln gemalt werden. Das fertige Objekt wird durch Anklicken von „**Keep**“ im Malwerkzeug abgeschlossen, dann wird das Malwerkzeug neu geöffnet für das nächste Objekt.

Natürlich können einmal gemalte Objekte auch später verändert werden. Dafür benutzt man die Smarties (siehe 2).

Außerdem können auch vorhandene Bilder importiert werden. Es kann ein Bild als Hintergrundbild definiert werden, dieses kann dann nicht mehr als Objekt angesprochen werden.

In diesem Projekt werden zwei Objekte gebraucht: ein Labyrinth, welches ausreichend groß sein sollte und ein beliebiges Objekt, das sich durch das Labyrinth bewegt, z.B. ein Käfer, ein Auto o.ä.

### Wie malt man einen Käfer?

1. **Navigator-Klappe öffnen, indem man auf den sichtbaren Griff klickt**
2. **Pinsel anklicken**
3. **Käfer malen, Einstellungen für Farbe, Stiftbreite usw. im Menü des Malwerkzeugs**
4. **„Keep“ im Malwerkzeug-Menü anklicken, wenn Käfer fertig ist**

## **2. Arbeiten mit den Objekten**

Wenn das Malwerkzeug beendet ist, hat man ein Objekt auf dem Bildschirm. Man merkt das daran, dass man dieses Objekt mit der Maus anfassen und herumschieben kann. Um in Squeak-Etoys mit den Objekten zu arbeiten, gibt es ein spezielles „Menü“, welches erscheint, wenn man mit der Maus ein Objekt anklickt und gleichzeitig die „Alt“-Taste auf der Tastatur drückt. Nun erscheinen einige runde Icons rund um das Objekt, die wegen ihres Erscheinungsbildes „Smarties“ genannt werden (im englischen heißt es Halo). Da alles ein Objekt ist, kann es auch vorkommen, dass man die Smarties für die „Welt“ bekommt (wenn man irgendwo auf den Bildschirm klickt statt auf ein Objekt) oder auch für die Navigatorklappe ....

### **Wie manipuliert man ein Objekt?**

- 5. Käfer mit der linken Maustaste anklicken und gleichzeitig ALT-Taste drücken (auf dem Mac die Apfel-Taste) oder**
- 6. die Maus über den Käfer bewegen und einen Moment warten, die Smarties erscheinen dann leicht transparent und erscheinen richtig, sobald man ein Icon auswählt oder**
- 7. Käfer mit der mittleren Maustaste anklicken**
- 8. für die gewünschte Manipulation das passende Icon anklicken. Eine Übersicht über die Icons und ihre Bedeutung ist in der beiliegenden Etoys-Kurzanleitung zu finden. Außerdem öffnet sich ein Hilfeballon, wenn man mit der Maus über einem Icon verweilt.**

### **3. Erstellen von Skripten für Objekte**

Die Interaktionsmöglichkeiten, die die Smarties eröffnen, sind sehr unterschiedlich. Da gibt es einfache Icons, um die Größe des Objektes zu verändern oder es zu drehen, es aufzuheben ,zu verschieben, zu kopieren, neu zu malen, die Farbe zu ändern oder es zu löschen. Nicht für jedes Objekt sind alle Icons verfügbar, weil manche Objekte bestimmte Manipulationen nicht erlauben.

Komplexere Funktionen stellen die Icons für das Menü und den Betrachter zur Verfügung. Im Menü sind viele Funktionen zu finden, die das Aussehen eines Objektes betreffen, die Zugang zur Programmierumgebung bieten und z.B. Geschwister des Objektes erzeugen können. Viele dieser Funktionen werden erst benötigt, wenn man sich tiefer mit Etoys auseinandersetzen will, einige sind aber auch gerade am Anfang nützlich. Mit dem Menüeintrag „Export“ ist es beispielsweise möglich, ein Objekt als Bild zu exportieren.

Das eigentliche Ziel in Etoys ist es aber, für die gemalten Objekte ein Verhalten zu beschreiben. Das erledigt man über Skripte. Das dafür benötigte Icon ist das „Auge“, das den Betrachter öffnet. Dieser Betrachter enthält eine Vielzahl von Kacheln, die je eine Befehlszeile enthalten. Damit werden die Skripte erzeugt.

#### **Wie erstellt man ein Skript?**

- 9. türkises Icon (mit dem Auge) des Käfers anklicken**
- 10. in der Titelzeile des Betrachters für den Käfer den Text „Zeichnung“ durch einen passenden Namen ersetzen, z.B. „Käfer“. Umlaute sind erlaubt!**
- 11. die Kachel mit dem gewünschten Befehl, z.B. „Käfer gehe vorwärts um 5“, mit der Maus auf den freien Bildschirm ziehen. Es wird automatisch ein Skriptrahmen erstellt.**
- 12. Durch Anklicken des gelben Ausrufezeichens wird das Skript genau einmal ausgeführt.**
- 13. Durch Anklicken der Uhr wird das Skript in den Zustand „laufend“ versetzt und nun ständig hintereinander ausgeführt.**
- 14. Wie oft ein Skript pro Sekunde aufgerufen wird, kann man einstellen, in dem man länger auf die Uhr klickt und dann die Tickrate einstellt.**

#### **4. Aufgabenstellung Beispielprojekt**

Das Beispielprojekt für diesen Workshop soll das Navigieren durch ein Labyrinth sein. Es soll sich ein Objekt, wie z.B. ein Käfer, durch ein Labyrinth bewegen. Dabei darf der Käfer die Grenzen des Labyrinthes nicht überschreiten. Für dieses Projekt sind viele Erweiterungen denkbar:

- der Käfer läuft nicht von allein sondern wird von einem Spieler gesteuert, das lässt sich bis zu einem PacMan-Spiel mit Punkten, Geistern usw. ausbauen
- verschiedene Labyrinth und Verhaltensweisen der Käfer werden ausprobiert und verglichen
- der Käfer wird während der Bewegung durch das Labyrinth animiert

Das grundlegende Verhalten des Käfers soll sein:

- er bewegt sich vorwärts
- stösst er an die Mauer des Labyrinthes, versucht er in eine andere Richtung weiterzugehen

## 5. Anleitung Beispielprojekt

Zum Nachlesen und weil es immer vorkommt, dass die Teilnehmer unterschiedliche Arbeitsgeschwindigkeiten haben, hier die Anleitung zum Nachlesen.

1. Einen Käfer malen.
2. Ein Labyrinth malen. Dabei die Größenverhältnisse beachten, der Käfer sollte zwischen den Mauern hindurchpassen, auch wenn er sich umdreht.
3. Käfer und Labyrinth auf dem Bildschirm in die gewünschte Position bringen.
4. Skript schreiben um den Käfer vorwärts gehen zu lassen. Dazu entweder aus dem Betrachter aus der Kategorie „Skripte“ eine Kachel „leeres Skript“ auf den Bildschirm legen, um es dann zu füllen oder direkt die „gehe vorwärts um 5“-Kachel auf den Bildschirm legen. Das Skript soll „krabbeln“ heißen.
5. Was passiert bisher? Der Käfer bewegt sich pro Skriptaufruf um 5 Pixel vorwärts. Wo ist vorwärts? Man holt die Smarties für den Käfer und achtet auf den grünen Pfeil in der Mitte. Dieser zeigt vorwärts. Man kann den Pfeil drehen, um eine andere Richtung als „vorwärts“ festzulegen. In der Kategorie „Einfach“ im Betrachter ändert sich der dazugehörige Wert für „Richtung“!
6. Skript schreiben, um den Käfer an der Mauer des Labyrinthes zu stoppen. Man kann den Test, ob der Käfer eine Mauer berührt, in das gleiche Skript schreiben, das auch schon das Vorwärtsgehen realisiert. Oder man schreibt ein neues Skript „testen“. Skripte können gleichzeitig, unabhängig voneinander ablaufen.
7. Um festzustellen, ob der Käfer eine Mauer berührt, braucht man eine Test-Kachel. Die bekommt man in einem Skript, wenn man das hellgelbe Rechteck links neben dem Löschen-Icon im Kopf eines Skriptes anklickt. Die Testkachel in das Skript legen.
8. Was soll getestet werden? In den Etoys sind die meisten Tests graphisch. Welche Tests genau möglich sind sieht man in der Kategorie „Tests“ im Betrachter des Käfers. Die Kachel „Käfers Farbe sieht“ mit der Maus in das Skript ziehen und hinter „Test“ im gelben Rechteck ablegen. Nun sind in der Kachel zwei farbige Rechtecke zu sehen. Man kann nun festlegen, welches Aufeinandertreffen von Farben getestet werden soll. Welche Farbe des Käfers soll welche andere Farbe sehen?
9. Mit der Maus auf das linke farbige Rechteck klicken, es erscheint eine Pipette. Mit dieser wählt man nun die Farbe am Käfer aus, die getestet werden soll, beispielweise die Beine. Dann klickt man auf das andere Rechteck und wählt die Farbe aus, bei deren Berührung etwas passieren soll. Also die Farbe vom Labyrinth. Spätestens hier zeigt sich, ob die Farben sinnvoll gewählt sind. Die Mauern des Labyrinthes sollten nicht kunterbunt sein!
10. Was soll nun passieren, wenn die Beine des Käfers die Mauer des Labyrinthes berühren? Er soll in eine andere Richtung weiterlaufen. Für die Änderung der Richtung wird eine „Käfer drehe dich um 5“-Kachel hinter das „JA“ im Testblock im Skript gelegt. Möglicherweise soll sich der Käfer aber weiter als nur um 5 Grad drehen? Dazu einfach den Zahlenwert ändern.
11. Das Skript bzw. beide Skripte starten und sehen, was passiert. Zum Starten von zwei Skripten gibt es im Lager eine „Skriptkontrolle“. Die kann man sich holen, indem man auf den Griff der Lagerklappe am unteren Bildschirmrand klickt. Man kann die Klappe auch öffnen, indem man am Griff zieht, das ist dann nützlich, wenn man Objekte aus der Klappe holen will, die weiter unten liegen. Die Skriptkontrolle wird auf den Bildschirm gelegt. Alle Skripte, die beim Klicken auf den grünen Knopf „go“ starten sollen, müssen jetzt in den Zustand „angehalten“ versetzt werden. Und dann „Go“.

12. Verhält sich der Käfer schon wie gewünscht? Nein? Dann kommt jetzt die Debugging-Phase.
  - Vielleicht dreht der Käfer sich im Kreis? Weil er, nachdem er sich gedreht hat, mit dem nächsten Bein die Mauer berührt? Dann sollte er vielleicht lieber erst mal ein Stück rückwärts gehen, ehe er sich dreht. Also eine „Käfer gehe vorwärts“-Kachel hinzufügen und einen negativen Wert eintragen.
  - Läuft der Käfer einfach über die Mauer? Weil die Mauer sehr schmal ist und er erst mal 5 Pixel vorwärts läuft und danach die Mauer schon nicht mehr berührt, sodass der Test mit Nein beantwortet wird? Dann sollte er lieber weniger Pixel vorwärts gehen, bevor er wieder testet, ob er die Mauer berührt.
  - Stört etwas anderes am Verhalten? Vielleicht hilft eine schrittweise Abarbeitung mit dem gelben Ausrufezeichen, um den Grund zu ermitteln. Die Skripte lassen sich auch mit den Schülern „spielen“, das kann man auch machen, bevor sie in Squeak geschrieben werden.
13. Wenn man den Weg nachvollziehen will, den der Käfer gegangen ist, kann man den Stift nach unten setzen. Es gibt eine Kategorie „Stifte“ im Betrachter und dort eine Kachel „Stift unten“. Die muss auf „wahr“ gestellt werden. Das kann man auch direkt im Betrachter machen, da die Einstellungen im Betrachter immer direkt den Einstellungen des Käfers entsprechen. (Betrachter von Käfer und graphisches Objekt Käfer sind zwei Repräsentationen desselben Datenbestandes!) Wenn man nun die Skripte startet und den Käfer laufen lässt, hinterlässt dieser auf dem Bildschirm eine „Spur“.
  - Hier hat man ein Turtle-Graphik-System, das eine Vielzahl von Anwendungsmöglichkeiten bietet. Gerade für jüngere Schüler ist z.B. die Aufgabe „Lass deinen Käfer einen Stern malen“ sehr interessant. Man kann geometrische Zusammenhänge erklären, Koordinaten und Winkel. Auch für ältere Schüler gibt es anspruchsvolle Aufgaben.
14. Wie kann ein Spieler den Käfer steuern? Die einfachste Möglichkeit ist, sich ein „Steuergerät“ zu malen, ähnlich einem Lenkrad fürs Auto. Wichtig für die Bedienung ist, dass man sich einen Referenzpunkt markiert.
15. Betrachter für das Lenkrad holen, Namen vergeben. Der Käfer soll jetzt auf die Bewegungen des Lenkrades reagieren, genaugenommen soll er sich so drehen, wie sich das Lenkrad dreht. Also kommt in das Skript des Käfers eine Kachel „Käfer drehe dich um“. Und nun müssen wir anstelle der „5“ den Wert aus dem Lenkrad übernehmen.
16. Aus dem Betrachter des Lenkrades nimmt man die Kachel „Lenkrads Richtung“ und schiebt sie so über „Käfer drehe dich um“, bis die „5“ grün markiert ist. Dann kann man sie fallen lassen und es steht dort „Käfer drehe dich um Lenkrads Richtung“.
17. Am besten schaltet man das Testskript jetzt aus bzw. erstellt zum selbersteuern jetzt ein neues Skript „steuern“, das eine Kachel zum Vorwärtsgehen und die o.g. zum Drehen enthält. Dieses Skript wird nun gestartet. Für das Lenkrad holt man sich die Smarties und benutzt das blaue unten links, um das Lenkrad zu drehen. Entsprechend dreht sich der Käfer!
18. Diese Steuerung ist nicht sehr handlich, besser geht es schon, wenn man die Übersetzung ändert, indem man beispielsweise schreibt „drehe dich um Lenkrads Richtung / 2“. Das erhält man, wenn man auf den kleinen grünen Pfeil am Ende der Kachel klickt und die entsprechende Operation auswählt.
19. Ist der Stift noch unten? Dann kann man sehen, wo man überall langgefahren ist bei dem Versuch, den Käfer aus dem Labyrinth zu steuern. Das lässt sich auch zu einem kleinen

Spiel weiter entwickeln. Punkte werden vergeben, je nachdem, wie lange man braucht. Und wenn man an die Mauern stößt, werden Punkte abgezogen. Wie kann man das umsetzen?

- Inzwischen sieht der Bildschirm wahrscheinlich ziemlich „bekritzelt“ aus. In der Kategorie „Stifte“ gibt es die Kachel „lösche Stiftspuren im Eigner“. Einmal auf das gelbe Ausrufezeichen klicken, dann ist alles wieder „sauber“. Oder man macht sich gleich einen Knopf dafür! Die Kachel auf den Bildschirm ziehen, dadurch erzeugt man ein neues Skript. Man sollte ihm einen sinnvollen Namen geben und dann im Kopf des Skriptes auf das umrandete Rechteck „Käfer“ klicken. Aus dem aufklappenden Menü wählt man „Knopf um dieses Skript auszuführen“. Den Knopf legt man irgendwohin und kann dann jederzeit den Bildschirm damit löschen. Das Skript wird hier immer nur genau einmal ausgeführt! So ist das auch eine gute Möglichkeit, um eine Art „Reset“-Skript zu schreiben, was zusätzlich noch den Käfer auf eine bestimmte Position bringt usw.
20. Um Punkte zu zählen braucht man eine Variable. Variablen erzeugt man, indem man im Betrachter des Käfers auf das kleine „v“ vor dem Namen „Käfer“ klickt. Im Dialogfenster gibt man als Namen für die Variable „Punkte“ an und es öffnet sich eine neue Kategorie mit der Variable. Standardmäßig hat die Variable einen numerischen Typ, den kann man aber bei Bedarf ändern im Menü vor der Kachel mit der Variable.
  21. Wir setzen die Punkte auf den Anfangswert „0“. Das kann man im Betrachter machen und auch in das Reset-Skript übernehmen. Dann fügen wir dem „steuern“-Skript eine Anweisung hinzu, die die Punkte in jedem Aufruf erhöht. Dazu nehmen wir die Kachel mit der Variable – am grünen Pfeil anfassen, um eine Zuweisungskachel zu erhalten! Und legen sie im Skript ab. Klick auf die grünen hoch-runter-Pfeile und Auswahl von „Punkte erhöhen um“, dann als Wert „1“ eintragen und fertig. Bei jedem Skriptaufruf wird jetzt die Punktezahl erhöht, da das Skript immer mit der gleichen Tickrate aufgerufen wird, steigen die Punkte also gleichmäßig mit der Zeit.
  22. Woher weiss ich, wie viele Punkte ich schon habe? Ich nehme mir aus dem Menü vor der Variable im Betrachter einen Beobachter und lege ihn auf den Bildschirm. Und nun das Skript starten.
  23. Reset-Skript vervollständigen, mit dem Testblock erkennen, wann das Ziel erreicht ist.