Dominik O’Kerwin

Dominik.okerwin@crealogix.com

Exposee

Dieses Dokument enthaltet alle Informationen   
über das momentan geplante Projekt.

Typing application

Fallstudie

Inhaltsverzeichnis

[Typing Application 2](#_Toc497417355)

[Management Zusammenfassung 2](#_Toc497417356)

[Wieso braucht es dieses Projekt? 2](#_Toc497417357)

[Was sind die Ziele? 2](#_Toc497417358)

[Welche Aufgabe gibt es? 2](#_Toc497417359)

[Systemspezifikationen 2](#_Toc497417360)

[Rollen 2](#_Toc497417361)

[Meilensteine 3](#_Toc497417362)

[Terminplan 3](#_Toc497417363)

[Point of no Return 4](#_Toc497417364)

[Risikomanagement 4](#_Toc497417365)

[Marketing 5](#_Toc497417366)

[Zielgruppe 5](#_Toc497417367)

[SEUSAG 5](#_Toc497417368)

[Systemgrenzen 5](#_Toc497417369)

[Einflussgrösse 5](#_Toc497417370)

[Unter-und Teilsysteme abgrenzen 5](#_Toc497417371)

[Schnittstellen ermitteln 5](#_Toc497417372)

[Analysieren 6](#_Toc497417373)

[Gemeinsamkeiten feststellen 6](#_Toc497417374)

[Würdigung 6](#_Toc497417375)

[Ist-Zustand 6](#_Toc497417376)

[SWOT Analyse 6](#_Toc497417377)

[Lösungsentwurf mit Varianten 7](#_Toc497417378)

[Variante 1 Java Applikation 7](#_Toc497417379)

[Variante 2 Webseite 7](#_Toc497417380)

[Variante 3 App 7](#_Toc497417381)

[Auswahl 8](#_Toc497417382)

[Diagramme 8](#_Toc497417383)

[Use Case 8](#_Toc497417384)

[Use Case Diagramm 8](#_Toc497417385)

[Hauptanwendungsfall 8](#_Toc497417386)

[Akteure 8](#_Toc497417387)

[Use Case Wörter und Sprachen hinzufügen 9](#_Toc497417388)

[Use Case Schwierigkeitsgrad auswählen 9](#_Toc497417389)

[Use Case Spiel spielen 10](#_Toc497417390)

[Use Case Definitionen verwalten 10](#_Toc497417391)

[Aktivitätsdiagramm 10](#_Toc497417392)

[Klassendiagramm 11](#_Toc497417393)

[Zielfindung 11](#_Toc497417394)

# Typing Application

## Management Zusammenfassung

### Wieso braucht es dieses Projekt?

Viele Applikationsentwickler die eine neue Programmiersprache erlernen, haben öfters das Problem, dass sie nicht alle richtigen Begriffen kennen für diese neue Sprache. Da gibt es noch Probleme für Personen die nicht so schnell im Schreiben sind oder auch für Personen die nicht so schnell diese Schlüsselbegriffe auswendig kennen. Aus diesem Grund braucht es eine Applikation, welche den Entwickler das gebrauchte Muskelgedächtnis antrainiert und die Schlüsselwörter beibringt. Dazu wird es auch zum Beispiel den Benutzer eines Mac Computers, auf welchem nicht alle Sonderzeichen auf der Tastatur angezeigt werden.

### Was sind die Ziele?

Die Ziele die wir für dieses Projekt definieren sind, dass für den Benutzer von dieser Applikation den Einstieg in einer neuen Programmiersprache erleichtert wird. Darunter wird es eine Library geben für die Schlüsselbegriffe, welche alle Definitionen beinhalten wird.

### Welche Aufgabe gibt es?

Damit dieser Auftrag erfüllt werden kann, müssen einige Aufgaben erreicht werden. Diese sind, dass man eine Möglichkeit hat das Muskelgedächtnis zu trainieren für die einzelnen Schlüsselbegriffe. Dazu kann der Benutzer die Schlüsselwörter für eine Programmiersprache editieren und auch welche hinzufügen. Da kann man auch eine eigene Programmiersprache hinzufügen.

# Systemspezifikationen

Diese Applikation wird in der Programmiersprache Java geschrieben. Dies erlaubt, dass man auf jedem Betriebssystem diese Anwendung verwenden kann, welches Java unterstütz. Dazu werden wir die Java Version 8 verwenden.

Das Projekt wird auf einem Repository sein, für welches Github verwendet wird. Dies erlaubt, dass wir keine Probleme haben beim miteinanderarbeiten an der Applikation.

# Rollen

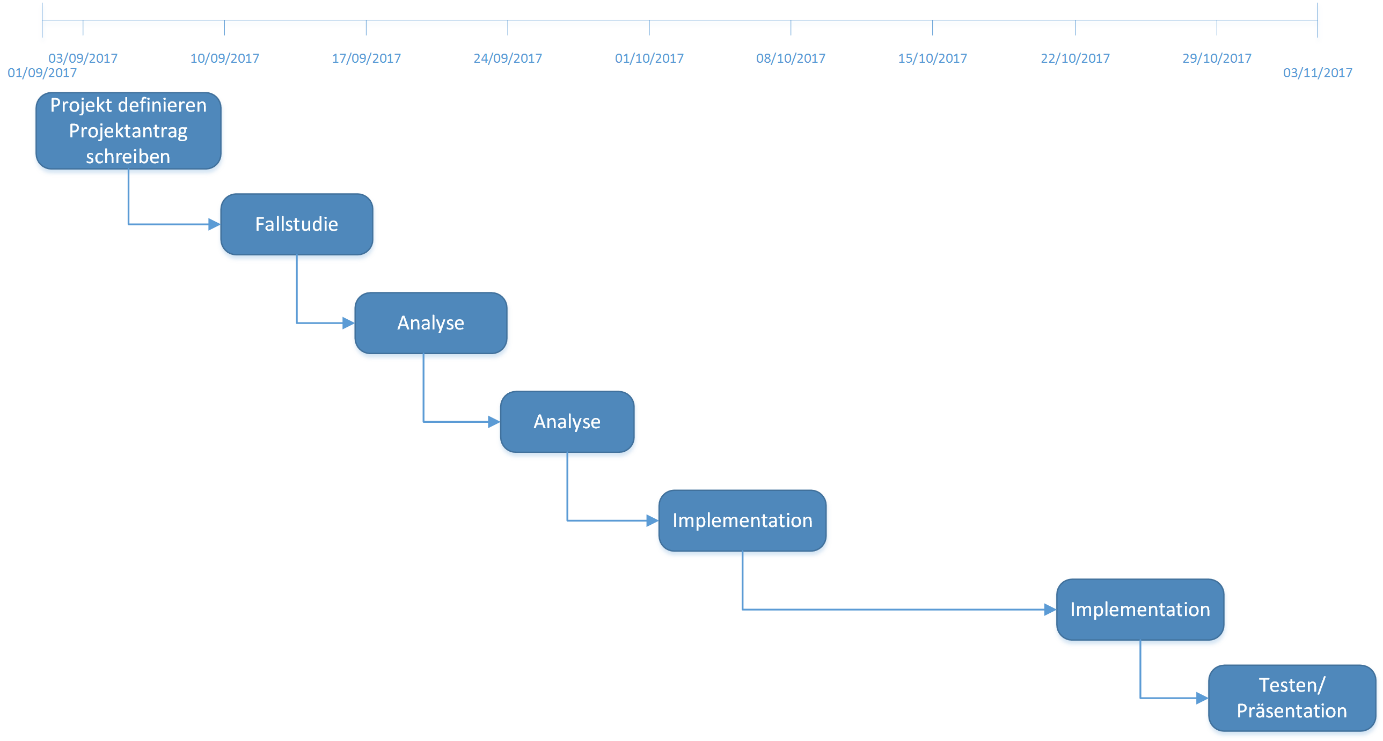
Die Rollen für dieses Projekt werden aufgeteilt zwischen den Lernenden Oliver Kälin und Dominik O’Kerwin. Somit müssen wir auch mehrere Rollen spielen und diese wären dies:

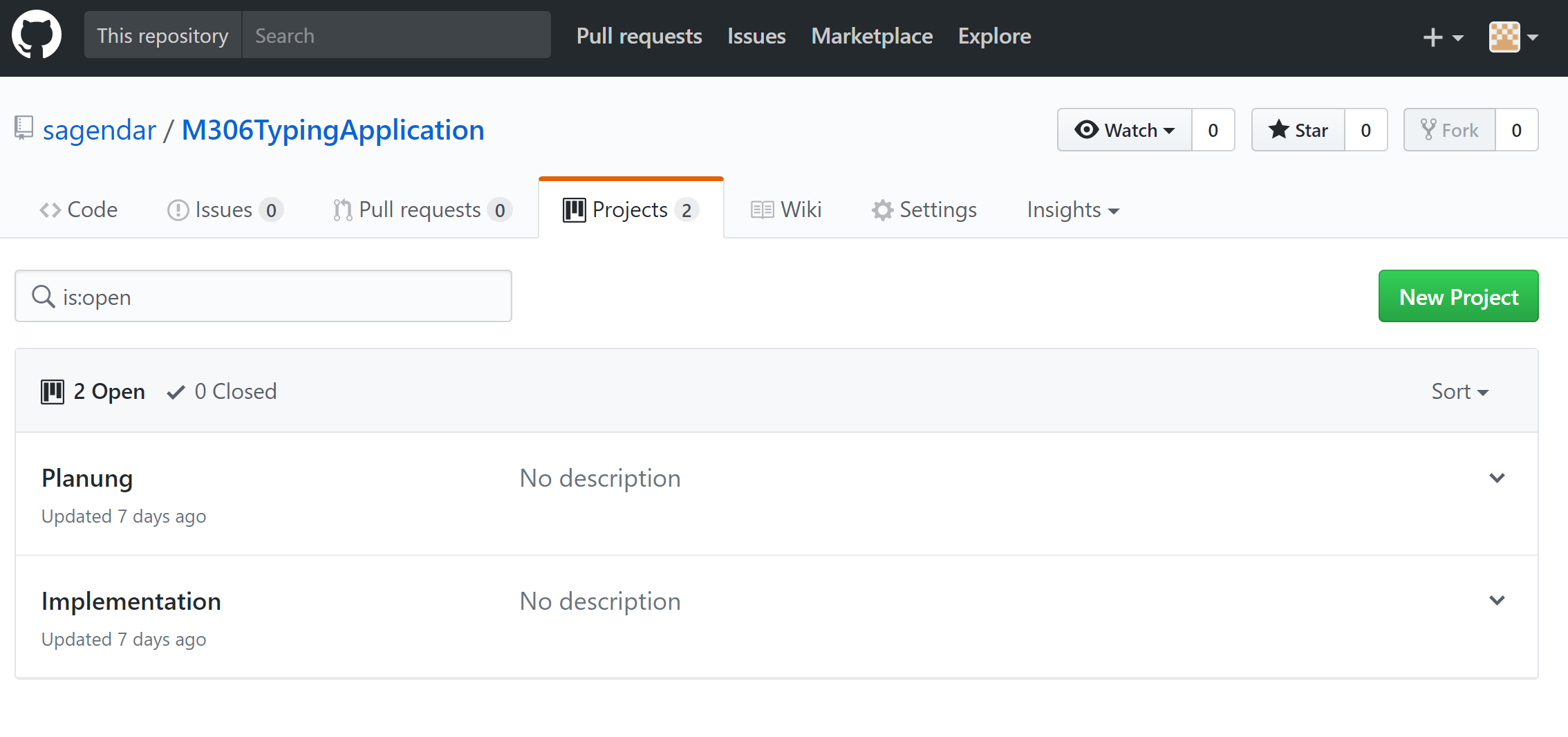
|  |  |  |
| --- | --- | --- |
| Rolle | Person | Beschreibung |
| Projektleiter | Dominik O’Kerwin | Wird dafür schauen, dass das Projekt am Schluss erfolgreich abgeschlossen wird. |
| Projekt-Mitarbeiter | Oliver Kälin | Unterstützt den Projektleiter bei den organisatorischen Aufgaben. |
| Planungsleiter | Dominik O’Kerwin | Diese Rolle beinhaltet, dass die Planung vom ganzen Projekt erstellt wird, damit das Projekt erfolgreich wird. |
| Planungsmitarbeiter | Oliver Kälin | Unterstützt den Planungsleiter bei diversen Aufgaben. |
| Programmierleiter | Oliver Kälin | Der Programmierleiter wird von der Planung aus, die Applikation erstellen und schauen, dass alles die Applikation erfolgreich abgeschlossen wird. |
| Programmieraffe | Dominik O’Kerwin | Unterstützt den Programmierleiter bei der |

# Meilensteine

## Terminplan

|  |  |  |
| --- | --- | --- |
| Datum | Ziele | Status |
| 01.09.2017 | * Projekt definieren * Projektantrag schreiben | Fertig |
| 08.09.2017 | * Persönliche Dokumentationen beginnen * Fallstudie beginnen * Rollenverteilung | Fertig |
| 15.09.2017 | * Fallstudie weiterarbeiten * Planung erstellen * Erste Diagramme erstellen | Fertig |
| 22.09.2017 | * Fallstudie weiterarbeiten * Erstes GUI erstellen | Fertig |
| 29.09.2017 | * Fallstudie weiterarbeiten * Erste Funktionalitäten sollten funktionieren | Fertig |
| 06.10.2017 | * Fallstudie beenden * Hauptfunktionalitäten implementiert | Fertig |
| 27.10.2017 | * Testen * Sekundäre Ziele implementieren | Fertig |
| 03.11.2017 | * Allfällige Bugs beheben |  |



Auf dem Github wird die genauere Planung dargestellt. Damit wird gemeint, dass wir für die zwei Hauptaufgaben Planung und Implementation verschiedene klein Ziele definiert haben. 

Innerhalb einer der Sektionen kann man dann alle Ziele sehen, welche gebraucht werden, um dieses Ziel zu erreichen. Dies würde normalerweise drei Status haben, welche sind «», «(In Progress)» und «(Done)».

## Point of no Return

Dieses Projekt wird ab dem Moment nicht mehr abgebrochen, wenn man die erste Implementation abgeschlossen hat. Das heisst, nachdem man das GUI erstellt hat und die ersten Funktionalitäten erfolgreich implementiert hat, wird man nicht mehr das Projekt abbrechen. Dies hat den Grund, dass man bis dahin sehr viel in dieses Projekt investiert hat und die Firma zu einem hohen Verlust erleiden würde, wenn

## Risikomanagement

Während der Entwicklungsphase kann es immer dazu kommen, dass einer der Mitarbeiter fehlen könnte oder sonstige Probleme auftauchen. Für diesen Zweck ist der letzte Tag eine Möglichkeit um noch allfällige Sachen noch zu implementieren. Dazu würde wenn bei einem Tag ein Teil der Ziele nicht erreicht wird, die auf das nächste Mal erledigt in bei der entsprechenden Person.

# Marketing

Diese Applikation wird nach der ersten Aufgaben Phase von diesem Projekt nicht gerade verkauft. Dies hat den Grund, dass diese Applikation noch nicht Marktbereit wäre für diesen Zweck. Aus diesem Grund wird nach dieser Phase eine Review Phase kommen. In dieser werden wir das Produkt weiterhin verbessern und eine kleine Gruppe von Testern einladen, welche diese Applikation testen und auch Verbesserungsvorschläge uns geben. Währenddessen kann man es auf Github gratis herunterladen. Dies führt dazu, dass wir auch noch Reviews von diversen Fremden Personen erhalten können und somit diese Applikation bekannter machen.

## Zielgruppe

Die Hauptzielgruppe für welche diese Applikation ist, sind die Lernende oder auch Berufseinsteiger in der Informatikbranche. Dies führt dazu, dass wir vor allem wenn das Produkt fertig sind, dass wir es an der Ausbildungsstätte für Informatiker verkaufen. Dazu werden noch normale IT Firmen kontaktiert um unser Produkt anzuwerben, damit sie es für sich verwenden können.

# SEUSAG

## Systemgrenzen

Die Grenzen für dieses System sind nur die Applikation selber. Diese heisst, sie wird eine Standalone Applikation, welche keinen Zugriff auf das Internet bräuchte. Damit man diese Applikation starten kann, muss man sie einfach herunterladen und Java auf dem Computer vorinstalliert haben. Dies erlaubt das jeder Person diese Anwendung verwenden kann.

## Einflussgrösse

Diese Applikation sollte möglichst kompakt sein und wenig Speicher auf einem Computer verwenden. Aus diesem Grund versuchen wir es so optimiert wie möglich zu Programmieren. Diese Applikation sollte dazu nur auf einem Computer/Laptop verwendbar sein, also muss es nicht als eine App oder als eine Webseite nachprogrammiert werden. Der Internetzugriff sollte so klein wie möglich gehalten werden, damit es einen Benutzer erlaubt, auch unterwegs diese Applikation zu verwenden oder wenn es einen Stromausfall gibt.

## Unter-und Teilsysteme abgrenzen

Das ganze System kann man insgesamt in 3 Teile unterteilen. Zuerst die Trainingseinheit oder auch das Spiel ist der essentielle Punkt von dieser Applikation.

Der zweite Punkt bezieht sich auf das Hinzufügen von Wörter und Programmiersprachen. Diese erlaubt, dass man immer mehr im Wörter lernen kann.

Beim letzten Punkt kommt die Definitionen von den erlernten Worten hinzu. Diese kann man, wenn man eine Internetverbindung hat, direkt nachschauen und kann somit wie ein kleines Definitionsbuch bei sich halten.

## Schnittstellen ermitteln

Die grösste Schnittstelle in dieser Applikation wäre, was der Benutzer eintippt und was daraus passiert wird. Wenn der Benutzer bei der Hauptfunktion der Applikation etwas eintippt, muss direkt überprüft werden, ob das eingetippte Wort richtig ist. Dazu kommt die Schnittstelle, wohin werden die Daten abgespeichert. Damit es sehr leicht zum Transportieren ist und nicht viel Platz aufbraucht, verwendet die Applikation eine CSV Datei, welche alle Informationen beinhaltet.

## Analysieren

Der Benutzer wird immer wieder mehrere Worte neu hinzufügen in seiner kleinen Applikation und somit wird sein Wissen und Training um einiges mächtiger. Damit man aber nun einen guten Nutzen daraus ziehen kann, werden die Benutzer gelegentlich gefragt, wenn sie mit dem Internetverbunden sind, ob wir ihre Informationen erhalten können. Diese beinhaltet, welche Worte sie verwenden, wie oft sie es falsch machen und so weiter. Dies erlaubt es uns, direkte Verbesserungen und ähnliche Sachen für die Benutzer zu erstellen.

## Gemeinsamkeiten feststellen

Ein Element welches in dieser Applikation überall vorkommen sind wohl die Worte und Kategorien, welcher der Benutzer für sich hinzufügt. Diese werden wirklich überall verwendet, in der Definition, beim Spiel und bei Verwalten von diesen. Wenn man diese Aufgabe sehr sorgfältig plant, kann man wenn man das richtig implementiert, sich sehr viel Zeit ersparen.

# Würdigung

## Ist-Zustand

Wenn man die heutige Jugend anschaut, wenn sie an einem Computer arbeitet, sieht man es ihnen wirklich an, dass sie mit dem Computer aufgewachsen sind. Sie sind es sich nicht ungewohnt Stundenlang an einem Computer zu sitzen und einfach irgendwas zu machen. Selbst die Erwachsenen sieht man es langsam an, dass sie sich an dieser neuen Ära des Computers dran gewöhnt haben. Dies alles macht es auch, dass wir alle relativ effektiv und schnell an einem Computer arbeiten können und nicht mehr ewig brauchen um etwas zu schreiben.

Was aber bei sehr vielen Personen wirklich fehlt ist, dass schreiben an einem Computer. Man kann sich ja denken, wenn sie schon so viel Zeit an einem Computer verbringen, dann würden sie ja sehr schnell sein beim Schreiben. Nur leider ist das nicht immer der Fall, da viele nicht die Zeit an einem Computer für Sinnvolle Sachen verwenden, sondern für diverse andere Sachen, welche dem Muskelgedächtnis fast nicht hilft. Selbst wenn sie relativ schnell sind im Schreibe, heisst es noch lange nicht, dass sie schnell alle Sachen schreiben können, welche ab und zu reines Muskeltraining ist.

## SWOT Analyse

|  |  |  |
| --- | --- | --- |
|  | Hilfreich  Um das Ziel zu erreichen | Schädlich  Um das Ziel zu erreichen |
| Intern  (Attribute der Firma) | **Stärken**  Durch diese Applikation können Mitarbeiter ihre Leistung steigern. Dies erlaubt auch, dass ein Mitarbeiter vielleicht länger an dieser Firma interessiert bleibt, da sie ihn versucht zu fördern und ihn auch mit Hilfsmittel unterstützt. | **Schwächen**  Bei minimalistischen Umsetzung oder auch Verwendung kann es einen Mitarbeiter sogar demotivieren und ihn sogar einen Grund geben die Firma zu verlassen. |
| Extern  (Attribute auf die Umgebung) | **Chancen**  Dieses Produkt kann man sehr schnell an einem grossen Markt verkaufen und die Nachfrage danach wäre nicht ganz klein. Dies erlaubt, dass die Firma einen sehr grossen Sieg erreichen könnten. | **Gefahren**  Die Firma steckt zu viele Ressourcen rein, aber der Markt interessiert sich am Schluss doch nicht so sehr, da die Applikation nicht so gut ist oder da es schon ein besseres Tool gibt. |

# Lösungsentwurf mit Varianten

Möglichkeiten der Technologien

Da diese Applikation auf einen weit verbreiteten Markt verbreitet werden soll, gibt es für diese Applikation 3 Technologien, die in Frage kommen. Diese sind eine Java Applikation, eine Webseite und ein App. Jeder diese Technologien bringt Vorteile und Nachteile, aber lass es uns mal genauer anschauen.

## Variante 1 Java Applikation

Mit einer Java Applikation gibt einen sehr grossen Vorteil, welches ist, dass man sie auf jedem Betriebssystem, welches Java unterstützt, die Applikation laufen lassen kann. Dazu ist diese Applikation schon auf einem Computer und man kann somit sein Muskelgedächtnis besser trainieren, da man an einer Tastatur arbeitet. Ein anderer Vorteil ist es, dass eine Java Applikation von allen Entwickler in diesem Team entwickelt werden können. Dazu sind diese relative gut darin und können so auf eine effiziente und gute Lösung kommen. Als letzter Vorteil kommt, dass wenn man diese Applikation erstellt, kann man sie unabhängig vom Internet verwenden und kann auch unterwegs diese Applikation verwenden, wenn man mal kein Internet hat.

Die Nachteile die es für Java gibt ist das man dieses Herunterladen muss und je nach wie gross das Produkt am Schluss sein wird, sehr viel Platz aufbrauchen kann. Ein anderes Problem wäre wohl, dass man wirklich Java haben muss und wenn ein Gerät nun dies nicht unterstützen würde, dann kann man die Applikation nicht mehr verwenden.

## Variante 2 Webseite

Der grosse Vorteil an dieser Lösung wäre, dass man es überall her verwenden kann und man braucht nur eine Internetverbindung. Dazu kann wenn man es vermarktet auch den Firmen einen eigenen Bereich erstellen lassen, in welcher sie ihre Mitarbeiter zugreifen lassen. Dies erlaubt auch einer Ausbildungsstätte, den aktuellen Stand von den Lernenden zu sehen und wie sie sich verbessern. Ein anderer Vorteil wäre natürlich, dass man so das Produkt leichter vermarkten kann, indem man direkt Demo Zugänge verschickt und erlaubt die Firmen, das Produkt zu testen.

Die Nachteile darin sind, dass man dafür eine konstante Internetverbindung braucht und dies wird man nicht immer haben. Dazu wird die Seite abhängig sein, auf unseren Server und falls es ein Problem gibt auf unserer Seite, wird man für eine längere Zeit nicht mehr Zugriff darauf haben. Sonst was noch ein negativer Punkt wäre, ist das es sehr teuer werden kann mit den hosten, da man einen Server konstant laufen lassen muss. Im Fall wo das Projekt scheitern sollte, würde man somit sehr viel Geld um sonst ausgegeben haben mit dem Hosting.

## Variante 3 App

Wenn man es als App vermarkten möchte, erhaltet man sicher eine grosse Bandweite von Benutzer. Diese sind vor allem interessiert an Programmieren. Indem man es zum Beispiel auf den Appstore oder Play Store hinzufügt, können viele Personen diese Applikation sehen. Dazu hat fast jede Person ihr Smartphone dabei und kann somit durchgehen aktiv lernen.

Was aber dir weniger hilft ist, dass man nicht wirklich das Muskelgedächtnis trainieren kann. Da man normalerweise an einer Tastatur arbeitet, ergibt sich hier weniger Chance dieses Wissen wirklich zu trainieren. Man würde dann die Branche wechseln müssen und es eher als einen Nachschlagewerk verwenden müssen, was aber nicht der Sinn dahinter ist.

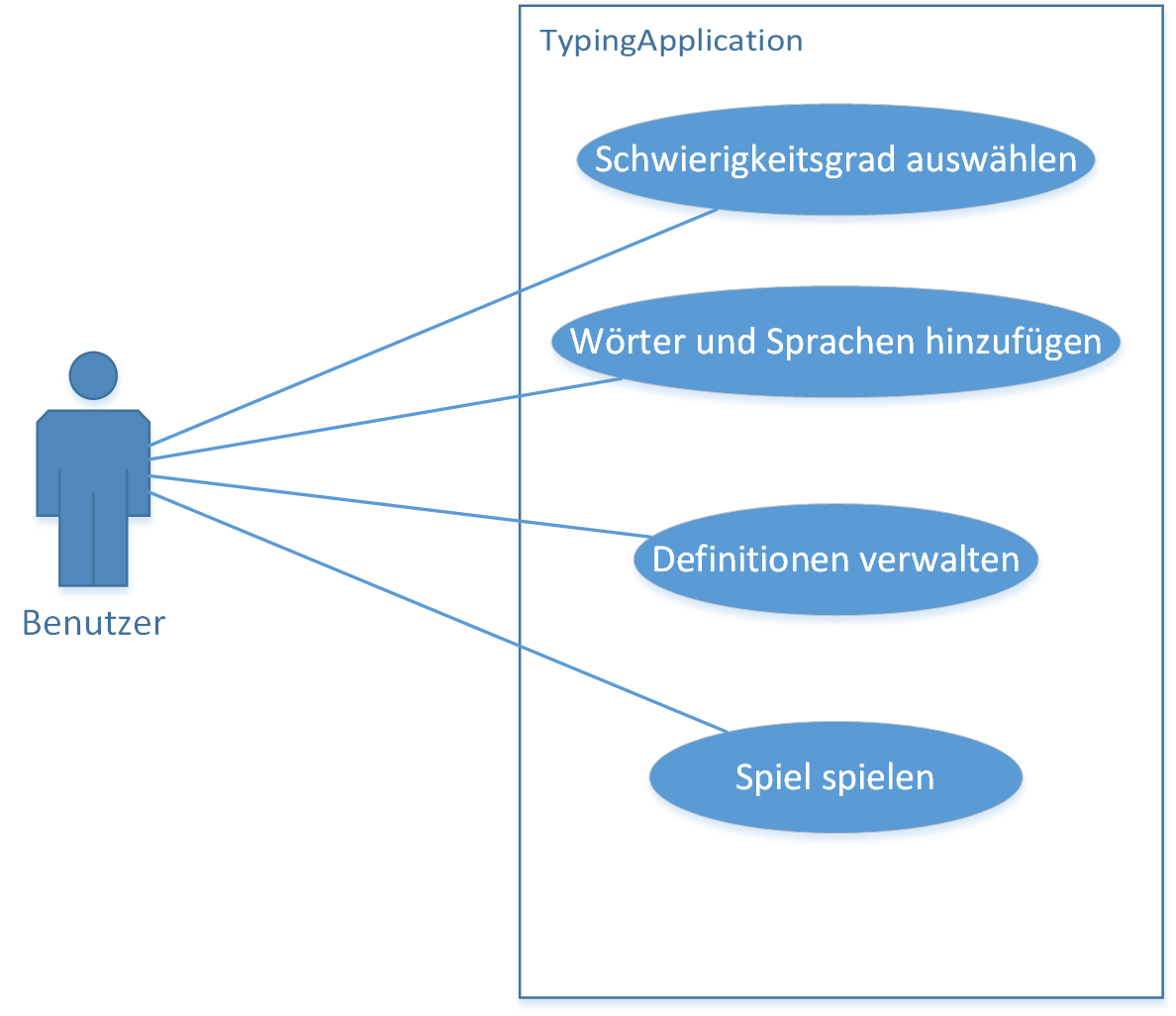
# Auswahl

Aus der Perspektive, dass nun sich eine schnelle und effiziente Lösung erstellt werden muss, haben wir uns für eine Java Applikation entschieden. Die Gründe dafür sind, dass unser Hauptprogrammierer in diesem Gebiet eher speziallisiert ist, als in den anderen. Dies ist aber nicht der Hauptgrund, aber immer noch ein Punkt den man sehr berücksichtigen mussten. Der andere Grund ist, dass es uns nicht die Idee gefallen hat, dass man eine Konstante Internetverbindung braucht und vor allem ist die Entscheidung für eine Webseite runtergefallen, da die Kosten und Aufwand im Gegensatz zu einer Java Applikation um einiges höher ausfallen. Dazu kommt das die Entwicklungsphase um einiges länger gedauert hätte, als wir überhaupt Zeit hätten. Der simple Grund wieso für uns die App Variante nicht in Frage gekommen ist, weil es dann den Zweck der Applikation komplett vernichten würde.

# Diagramme

## Use Case

### Use Case Diagramm



### Hauptanwendungsfall

Der Hauptanwendungsfall von dieser Applikation ist, dass der Benutzer die Applikation startet und zuerst auf dem Menu kommt. Nachdem wird er im Spielmenu gehen, eine Programmiersprache und Schwierigkeitsgrad auswählen und die Übung machen. Da wird eine Weile trainieren und sein Muskelgedächtnis verbessern und verlässt zum Schluss die Anwendung.

### Akteure

|  |  |
| --- | --- |
| Akteur | Rolle |
| Benutzer | Der Benutzer wird der einzige Akteur für diese Applikation sein. Dieser wird die Applikation verwenden und somit sein Muskelgedächtnis verbessern. |

### Use Case Wörter und Sprachen hinzufügen

|  |  |
| --- | --- |
| Name | Wörter und Sprachen hinzufügen |
| Beschreibung | Der Benutzer möchte mehr Wörter lernen von einer bestimmten Programmiersprache. Aus diesem Grund fügt er eine Kategorie hinzu und die entsprechenden Wörter. |
| Beteiligte Akteure | Benutzer |
| Aktion | Es werden mehr Wörter und Kategorien hinzugefügt |
| Auslöser | Wenn der Benutzer vom Hauptmenu auf Wörter und Sprachen verwalten geht. |
| Vorbedingung | Die Applikation muss laufen und man muss im Hauptmenu sein. |
| Ablauf | |  |  | | --- | --- | | Wörter und neue Kategorie erstellen | Wörter in schon einer vorhandenen Kategorie hinzufügen | | Zum Wörter und Kategorien Seite gehen | | | Erstellen Knopf klicken | | | Wort hinzufügen | | | Neue Kategorienamen hinzufügen | Vorhandenen Kategorienamen hinschreiben | | Erstellen klicken | | |
| Ergebnis | Das System besitzt nun mehr Wörter die jemand erlernen kann. |

### Use Case Schwierigkeitsgrad auswählen

|  |  |
| --- | --- |
| Name | Schwierigkeitsgrad auswählen |
| Beschreibung | Der Benutzer wählt einen Schwierigkeitsgrad aus |
| Beteiligte Akteure | Benutzer |
| Aktion | Je nach Schwierigkeitsgrad werden die Worte umso schneller fallen und bewirkt das der Benutzer sich mehr herausfordert. |
| Auslöser | Wenn der Benutzer vom Hauptmenu auf TypingGame klickt. |
| Vorbedingung | Die Applikation muss laufen und man muss im Hauptmenu sein. |
| Ablauf | |  |  |  | | --- | --- | --- | | Schwierigkeit «Leicht» | Schwierigkeit «Mittel» | Schwierigkeit «Schwer» | | Zum Wörter und Kategorien Seite gehen | | | | Sprache auswählen | | | | Schwierigkeitsgrad Leicht auswählen | Schwierigkeitsgrad Mittel auswählen | Schwierigkeitsgrad Schwer auswählen | | Wörter fallen alle 2 Sekunden | Wörter fallen jede Sekunde | Wörter fallen alle 0.5 Sekunden | |
| Ergebnis | Der Benutzer hat nun ein Tempo, mit welchem er gut das ganze erlernen kann. |

### Use Case Spiel spielen

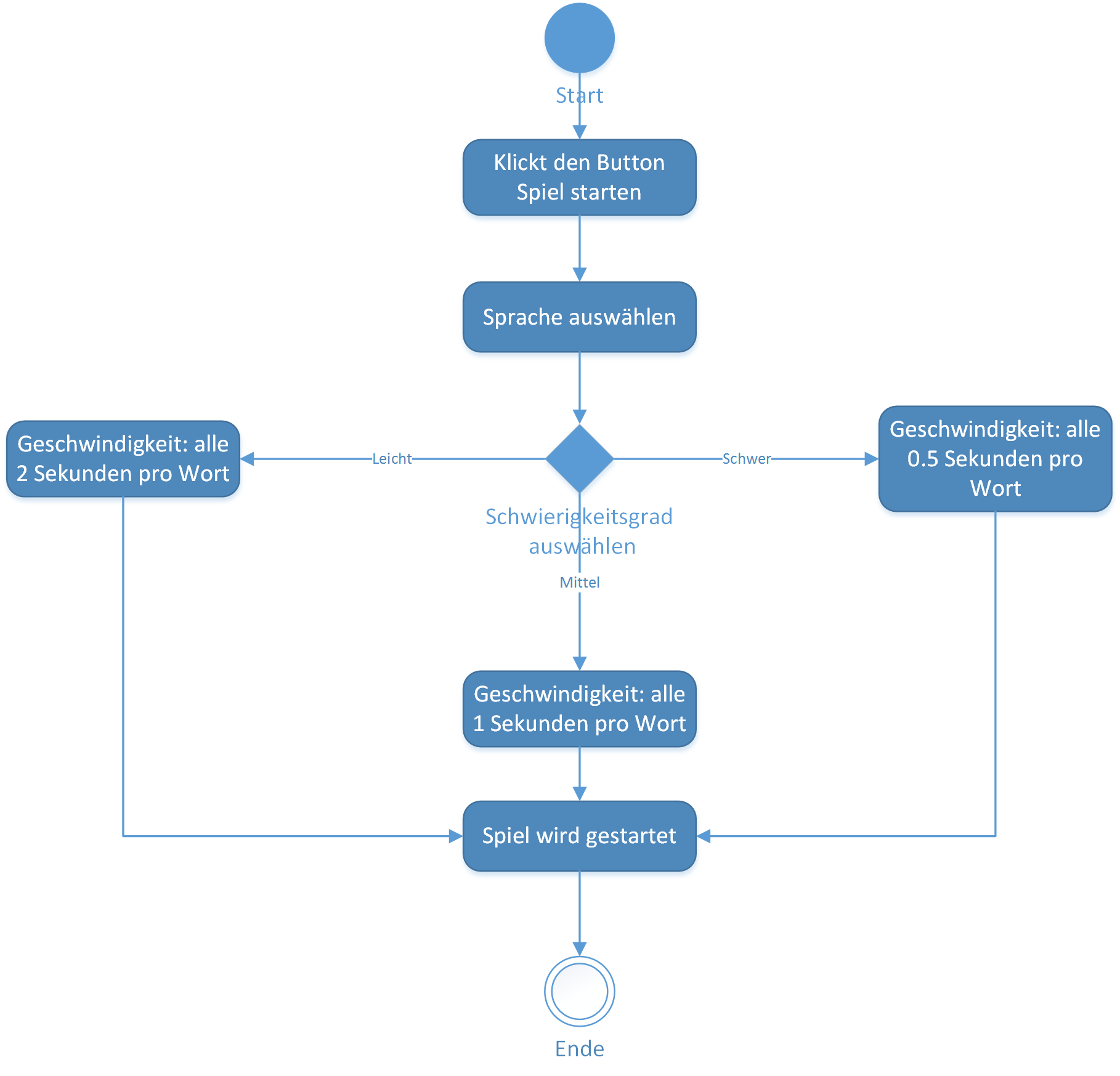
|  |  |
| --- | --- |
| Name | Spiel spielen |
| Beschreibung | Der Benutzer trainiert nun sein Muskelgedächtnis |
| Beteiligte Akteure | Benutzer |
| Aktion | Die Wörter werden herunterfallen und man kann die Wörter schreiben, damit sie verschwinden. |
| Auslöser | Wenn der Benutzer den Use Case «Schwierigkeitsgrad auswählen» erfolgreich durchzogen hat. |
| Vorbedingung | Use Case «Schwierigkeitsgrad auswählen» wurde beendet |
| Ablauf | |  |  | | --- | --- | | Erfolgreiches schreiben | Bei Fehler | | Wörter fallen hinunter | | | Benutzer tippt die Wörter hinein | Benutzer versucht die Wörter zu schreiben | | Wort verschwindet nach richtigen Schreiben | Wort fällt komplett nach unten | | Benutzer erhaltet einen Punkt für seinen Score | Der Score vom Benutzer wird sich um 1 senken | |
| Ergebnis | Der Benutzer wird nun schneller im Schreiben von diesen Schlüsselworten. |

### Use Case Definitionen verwalten

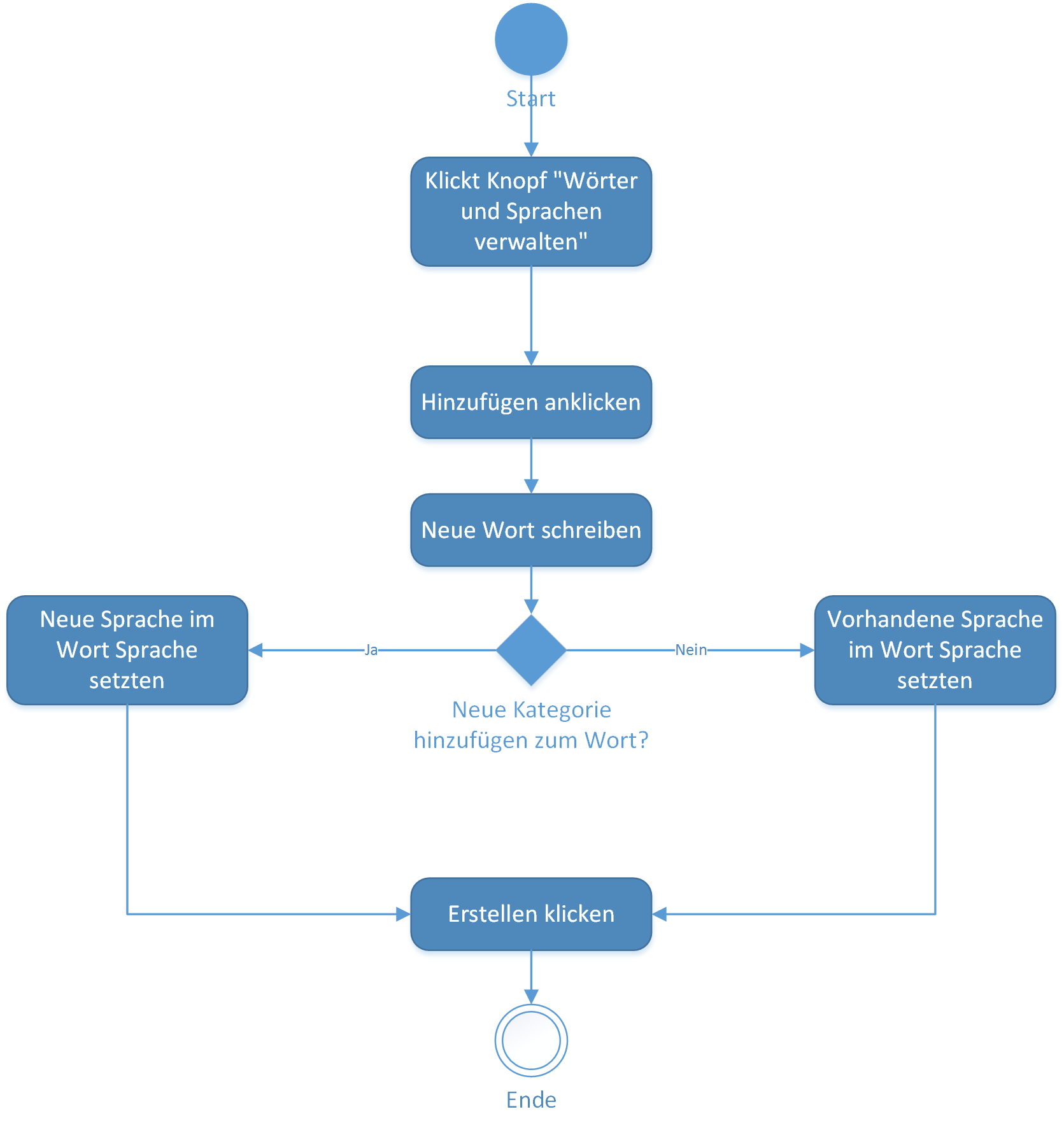
|  |  |
| --- | --- |
| Name | Spiel spielen |
| Beschreibung | Der Benutzer kann zu den vorhandenen Worten im System ihren Sinn nachschauen. Diese sind mit Hilfe von einem Link dargestellt. |
| Beteiligte Akteure | Benutzer |
| Aktion | Der Benutzer kann jeden Begriff nachschauen, welcher er hinzugefügt hat. |
| Auslöser | Wenn der Benutzer auf Definition im Hauptmenu klickt. |
| Vorbedingung | Applikation muss laufen |
| Ablauf | |  |  | | --- | --- | | Vorhandene Definition | Hinzufügen von Definitionen | | Benutzer klickt auf Definitionen | | | Der Benutzer wählt eine Sprache aus | | | Der Benutzer wählt ein Wort aus | | | Er klickt auf den link und sieht genau, was dieser Begriff bedeutet. | Der Benutzer verlinkt nun eine Seite für das entsprechende Wort | | - | Er speichert nun die Definition ab | |
| Ergebnis | Der Benutzer wird nun schneller im Schreiben von diesen Schlüsselworten. |

## Aktivitätsdiagramm

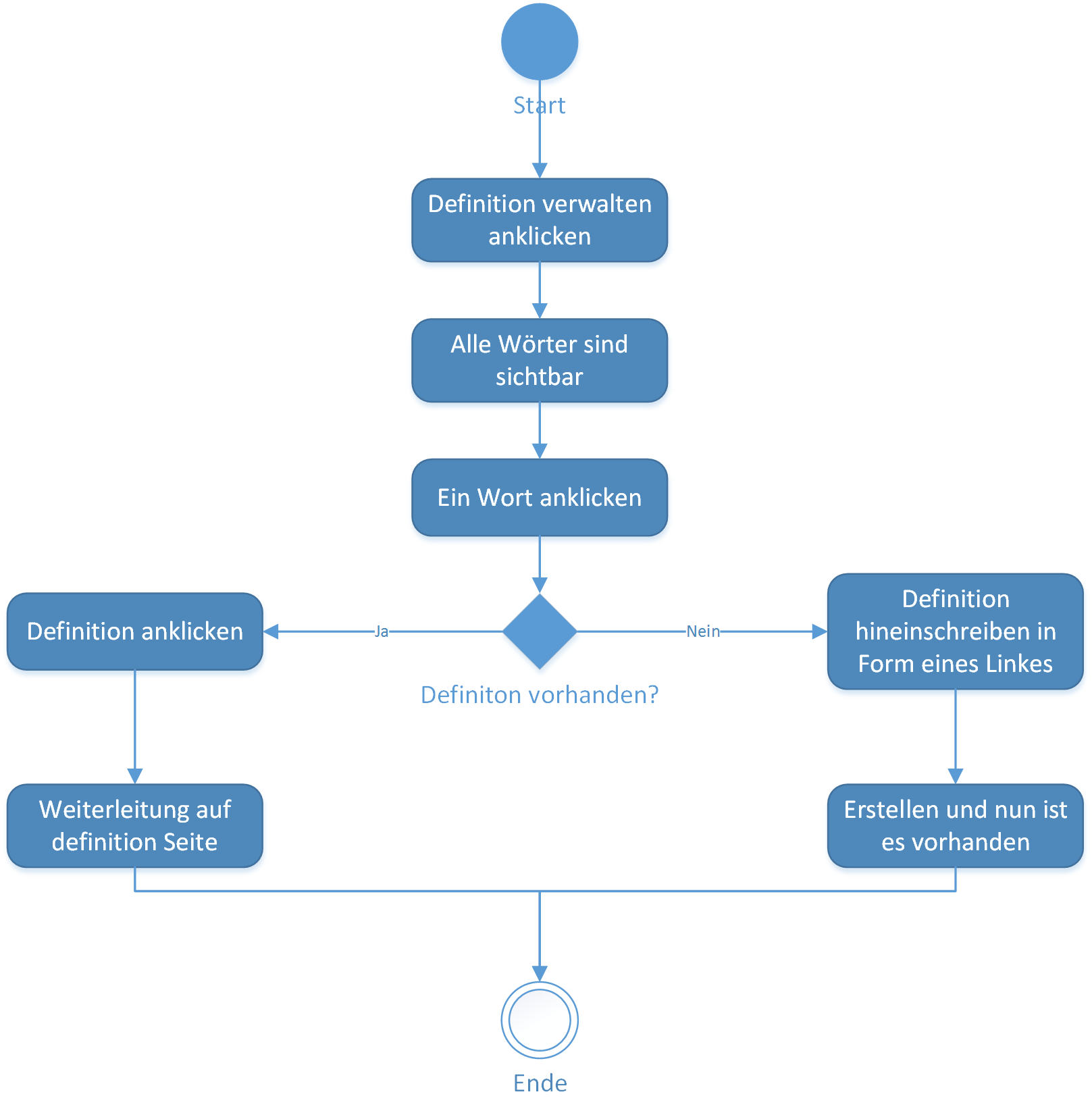
### Schwierigkeitsgrad auswählen



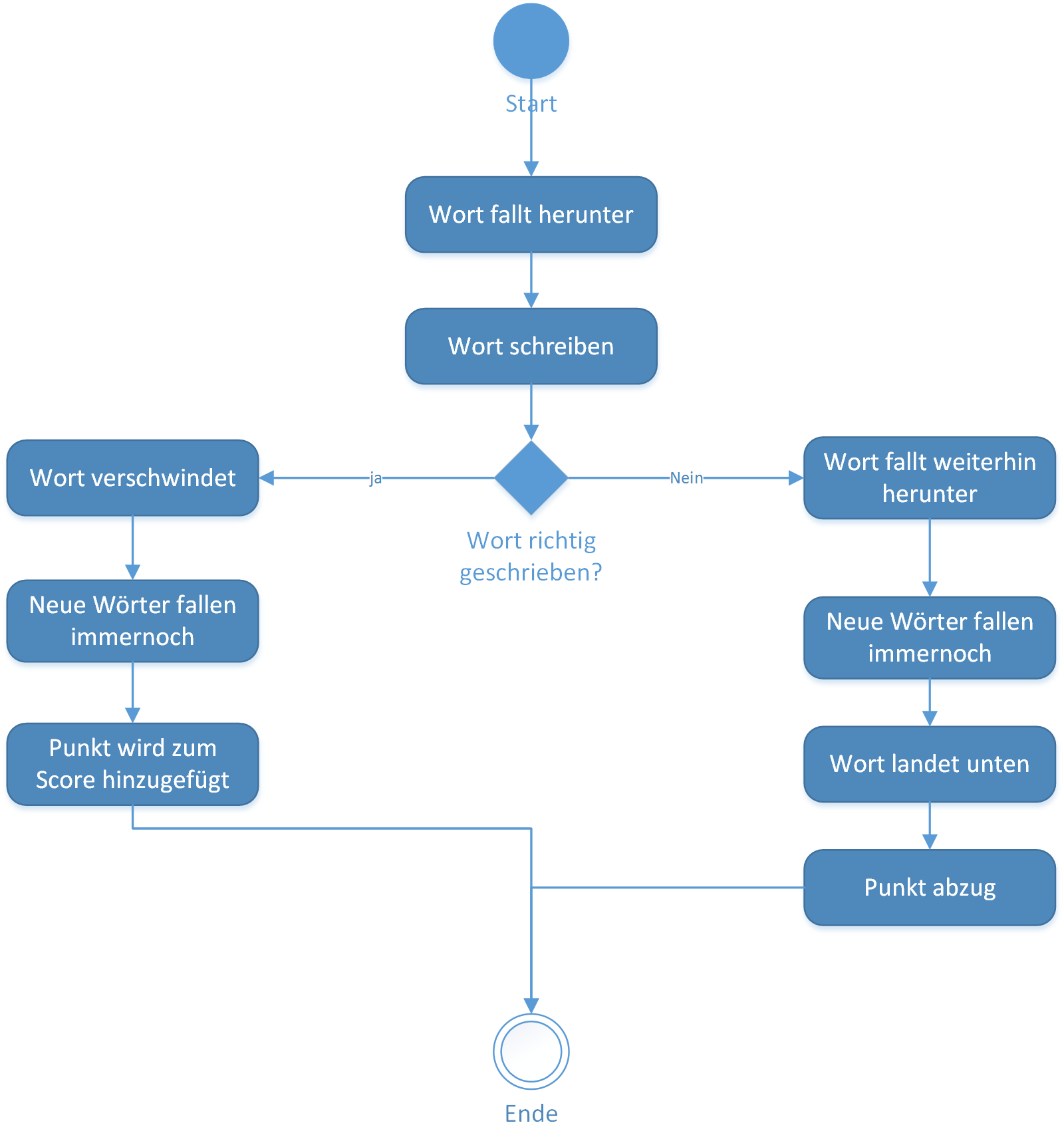
### Wörter und Sprachen verwalten



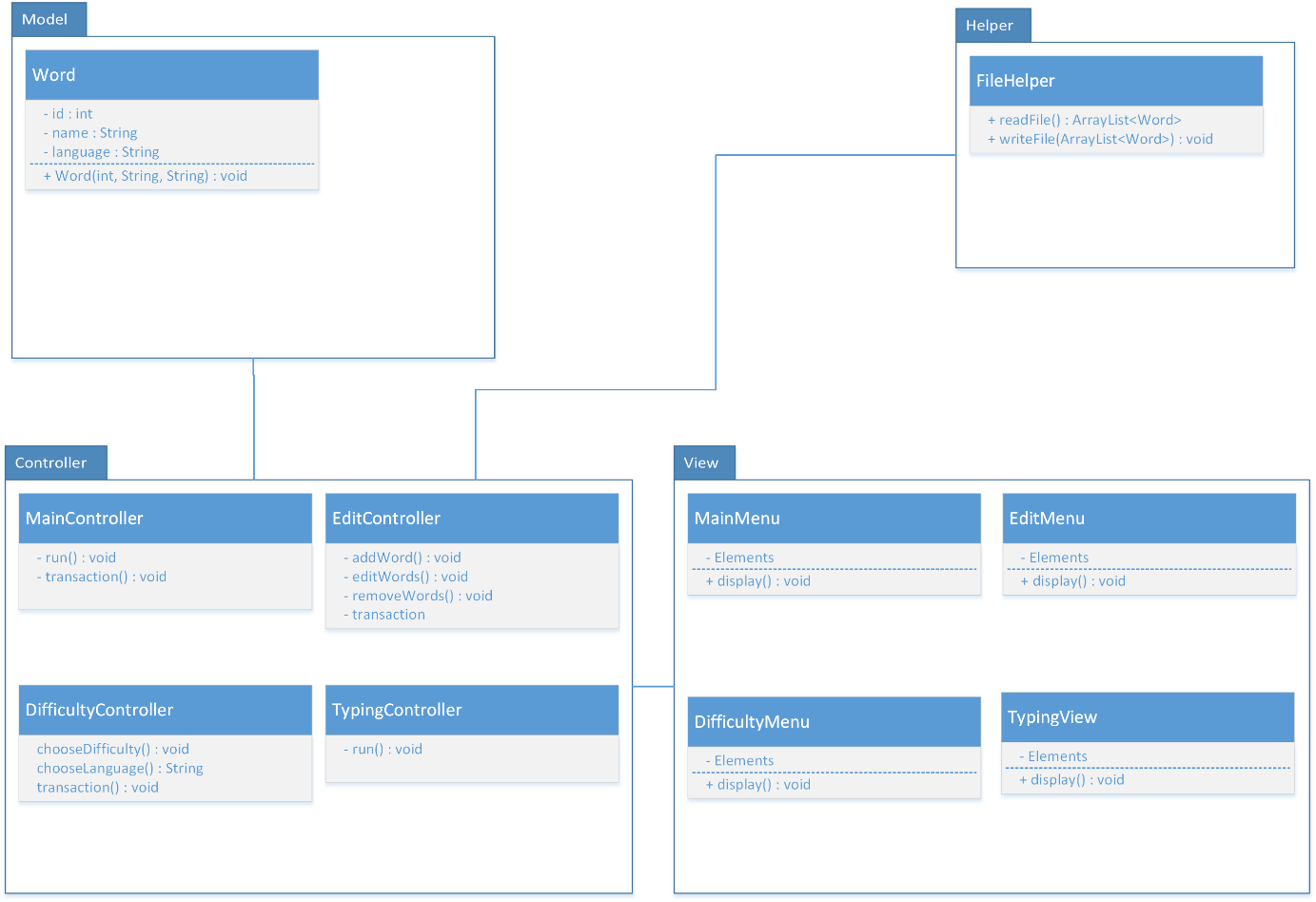
### Definition verwalten



### Spiel spielen



## Klassendiagramm



# Zielfindung

Dieses Projekt wird ein Hauptziel haben und dieses wäre ein Übungsprogramm für Entwickler zu entwickeln, welches den Einstieg in einer neuen Programmiersprache erleichtert. Andere Ziele welche definiert wurden werden hier nun aufgezeigt.

|  |  |  |
| --- | --- | --- |
| Ziel | Beschreibung | Gewichtung |
| Fallende Worte | Die Worte welcher der Benutzer sich zur Übung gespeichert hat, werden eine Animation haben, wo sie von oben nach unten fallen. Dazu soll wenn der Benutzer das richtige Wort eingibt, dass Wort verschwinden. | 40 |
| Punktestand | Wenn man ein Wort richtig geschrieben hat, wird der Benutzer einen Punkt erhalten und es wird dargestellt. | 5 |
| Schwierigkeitsgrade | Zu Beginn kann der Benutzer für jede Programmiersprache einen Schwierigkeitsgrad auswählen, welche leicht, mittel und schwer ist.  Bei leicht werden eher die kürzeren Wörter kommen und relativ langsam.  Bei mittel werden alle Wörter überprüft.  Bei schwer werden alle Wörter vorkommen und dazu wird sich die Geschwindigkeit der fallenden Wörter erhöhen. | 10 |
| Eigene Wörter und Programmiersprachen hinzufügen | Der Benutzer soll in der Lage sein, eigene Wörter und Programmiersprachen hinzuzufügen. | 15 |
| Wiki | Es gibt eine Sektion, wo man die verschiedenen Wörter die man hinzufügt anschauen kann und dazu ihren Nutzen anschauen. Dies kann vom Benutzer selber geschrieben werden oder auch nur eine Verlinkung zu der richtigen Seite sein. | 5 |
| Benutzerfreundlichkeit | Für den Benutzer soll das Interface sehr ansprechend sein und alles eine klare Navigation haben. | 20 |
| Unterstützung des Lernfortschrittes | Viele Worte wird man zu Beginn falsch schreiben oder einfach zu langsam sein. Für diesen Grund wird es eine Liste geben für jede Sektion, welche darstellt wie oft hat man ein Wort falsch geschrieben. Diese Worte werden auch öfters im Laufe des Spieles vorkommen, damit man sie öfters schreiben kann. | 10 |