



Objective C

Dominik Sacha
Tobias Schmid
Thomas Heidrich

Konstanz, 10.3.11

PROJEKTBERICHT

Zusammenfassung (Abstract)

Thema: Anagrammspiel - iDictAttack

Studenten: Dominik Sacha
Tobias Schmid
Thomas Heidrich

Betreuer: Julian Hirt

Abgabedatum: 12.03.2011

Schlagnworte: Spiel, Anagramm, Apple, Wörterbuch

In diesem Projekt wird ein Spiel entwickelt, in welchem Nutzer Anagramme in lexikografisch korrekte Wörter überführen. Schwierigkeitsgrade, Spielabläufe, Spielerbewertungsmaßstäbe und Eingabemöglichkeiten werden in diesem Bericht analysiert und bewertet. Weiter werden grundlegende Designentscheidungen getroffen und die Umsetzung dokumentiert.

Einleitung.....	4
Aaammngr (Anagramm).....	4
Grundlegendes Spielkonzept.....	4
Zielgruppe.....	6
Problemstellung.....	6
Eingabe des Lösungswortes.....	6
Hilfe.....	7
Vergleichsmöglichkeit von Erfolgen.....	7
Schwierigkeitsgrade.....	8
Trainingsmodus.....	8
Gegen die Zeit.....	10
Benutzeroberflächen.....	12
Menüstruktur.....	12
Entwürfe.....	14
Technische Konzepte.....	17
Komponenten.....	17
Kommunikation.....	18
Persistenz.....	19
Logik.....	20
Oberflächen.....	22
Fazit.....	22
Weblinks.....	23

Einleitung

Computerspiele werden heutzutage auf unterschiedlichsten Geräten wie beispielsweise Desktoprechnern, Laptops, Handhelds, Spielekonsolen und Mobiltelefonen gespielt. Zwischen diesen Geräten bestehen teilweise gravierende bauliche Unterschiede. Handhelds und Mobiltelefone werden z.B. fast ausschließlich von einem Akku mit Energie versorgt und besitzen einen relativ kleinen Bildschirm. Desktoprechner, Laptops und Spielekonsolen werden während des Spielens in der Regel mit Netzstrom versorgt und haben für gewöhnlich vergleichsweise große Bildschirme.

Eines haben jedoch alle gemeinsam. Sie ermöglichen es dem Nutzer, sich die Zeit zu vertreiben. Je nach Gerätetyp werden von den Nutzern verschiedene Anforderungen gestellt. Für ein Spiel auf einem Mobiltelefon ist beispielsweise die Kürze der Ladezeit um ein vielfaches wichtiger, als auf einem PC. Es ist einfach unpraktisch, wenn man mit seinem Handy an einer Bushaltestelle steht und drei Minuten auf ein Spiel warten muss.

Weiter haben die verschiedenen Spielegenres für gewöhnlich einen Gerätetyp, wo sie bevorzugt gespielt werden. Spiele mit vielen schnellen und präzisen Interaktionen wie Taktikshooter und Rollenspiele werden bevorzugt am PC gespielt, weil die Nutzer dort die notwendigen Eingabegeräte zur Verfügung haben. Gelegenheitsspiele wie Tetris [TETR] oder Snake [SNAK] werden hingegen bevorzugt auf Mobilgeräten wie Handys und Handhelds gespielt.

Das in diesem Projekt erstellte Programm zielt auf das Genre der Gelegenheitsspiele ab und es stehen nur die beiden folgenden Punkte fest. Das Spiel ist für das iPhone [IPHO] zu entwickeln und es soll um Anagramme gehen.

Der Name des Spiels lautet *iDictAttack*. Der Name besteht aus drei Teilen. Das "i" ist im Applekontext, wo das iPhone dazu gehört, weit verbreitet. Es soll kennzeichnen, dass es auf Appleggeräten ausgeführt werden kann. Die Wörter "DictAttack" beziehen sich das englische "dictionary attack" was so viel bedeutet wie "Wörterbuchangriff". Dies wurde gewählt, weil in diesem Spiel ein Wörterbuch zum Einsatz kommt, gegen welches ein Spieler spielt. *iDictAttack* hat nichts mit dem Wörterbuchangriff aus dem Bereich der Kryptoanalyse zutun, wo Passwörter mithilfe von Wortlisten entschlüsselt werden.

Dieses Kapitel erklärt ausführlich, was sich hinter dem Begriff Anagramm verbirgt, welches Spielkonzept angedacht ist und welche potentiellen Nutzer infrage kommen.

Aaammngr (Anagramm)

Ein Anagramm ist ein Wort, bei dem durch Neuordnung der Buchstabenreihenfolge ein neues lexikographisch korrektes Wort entsteht. Je nach Länge des Anagramms kann es unterschiedlich schwer sein, ein korrektes Wort zu finden.

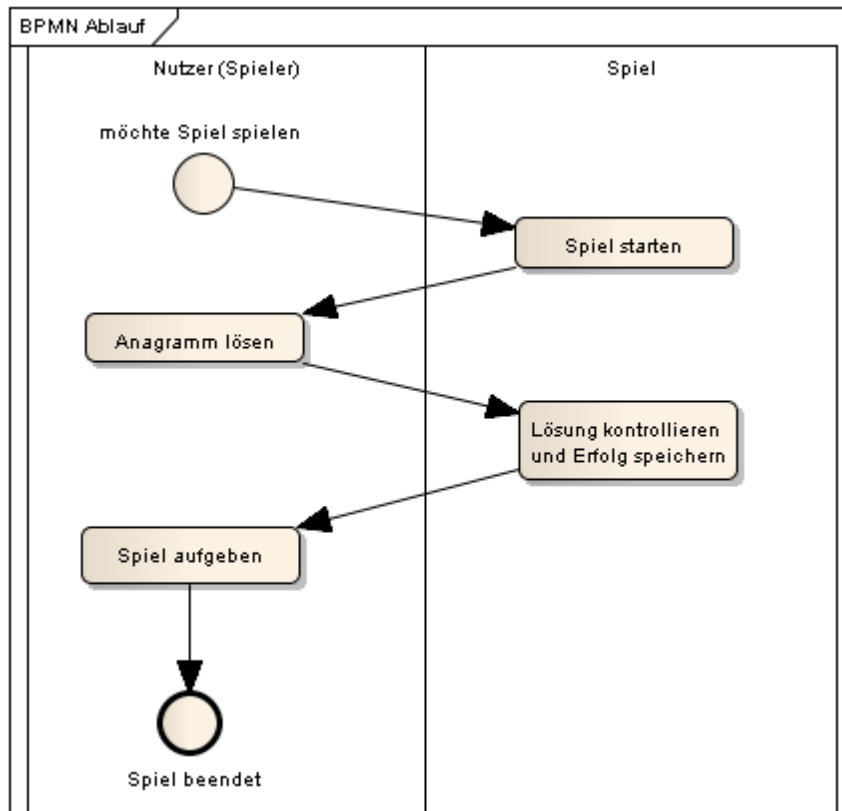
Beispielsweise ist das Anagramm *roetns* kein lexikographisch korrektes Wort der deutschen Sprache. Aus ihm können z.B. die korrekten Wörter *ortens*, *ostern*, *rosten*, *sorten*, *tenors* und *tensor* gebildet werden. Das Anagramm hat folglich mehrere Lösungen.

Das Anagramm *eirihshaenfdcfr* hat stattdessen nur eine Lösung. Sie lautet "friedrichshafen".

Die Lösung bzw. Lösungen eines Anagramms zu finden, stellt für einen Menschen eine gewisse Herausforderung dar, welche nun im Kontext eines Computerspiels umgesetzt wird.

Grundlegendes Spielkonzept

Das Spiel hat den Charakter eines Puzzle, da etwas in die richtige Reihenfolge gebracht werden muss, damit es gelöst wird.



1:Ablauf

Im folgenden Diagramm wird der grundlegende Spielablauf vereinfacht dargestellt. Dabei möchte ein Nutzer das Spiel spielen. Hierfür startet er das Spiel, welches ihm daraufhin ein Anagramm ausgibt. Der Nutzer gibt eine Lösung ein, das Spiel kontrolliert die Lösung und speichert, Daten zur Bewertung des Spielers. Danach beendet der Spieler das Spiel.

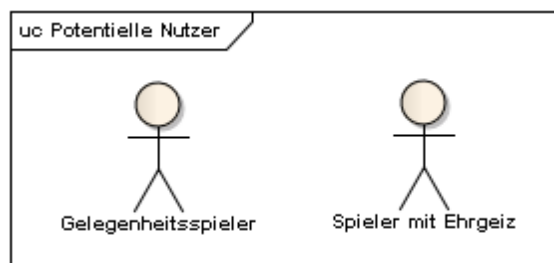
Der Prozess vom Lösen des Anagramms und der Kontrolle der Lösung kann mehrfach nacheinander erfolgen und ist im Diagramm nur einmal illustriert.

Diagrammelement	Beschreibung
Anagramm lösen	Das Anagramm wird vom in ein lexikographisch korrektes Wort überführt.
Lösung kontrollieren und Erfolg speichern	Die Lösung wird kontrolliert und es werden abhängig von der Richtigkeit verschiedene Ausgaben gemacht. Weiter wird der Erfolg bzw. Misserfolg gespeichert, um eine Bewertung der Spielleistung durchführen zu können.
Spiel aufgeben	Sobald der Spieler nicht mehr spielen möchte, gibt er das Spiel. Weiter kann er vom Spiel eine Aufgabe erzwungen werden, wenn der Spieler bestimmte Anforderungen vom Spiel nicht erfüllt.

	Der Spieler kann seinen Namen eingeben. Dieser wird zusammen mit seinem erreichten Erfolg in einer Bestenliste gespeichert.
Spiel starten	Das Spiel wird gestartet und danach gibt es ein Anagramm aus, welches lexikographisch inkorrekt ist.
Spiel beendet	Das Spiel ist beendet und kann jederzeit wieder gestartet werden.
möchte Spiel spielen	Der Ausgangspunkt ist, dass ein Spiel gespielt werden soll.

Zielgruppe

Um Software erfolgreich entwickeln zu können, muss festgelegt sein, welche Zielgruppe(n) angesprochen werden sollen.



2:Potentielle Nutzer

In diesem Diagramm sind alle potentiellen Nutzer abgebildet. Eine kurze Beschreibung erläutert, was hinter deren Bezeichnung steckt.

Diagrammelement	Beschreibung
Gelegenheitsspieler	Diese Art Spieler möchte sich zur Überbrückung von Wartezeiten dem Spiel widmen. Dabei sind dem Spieler niedrige Herausforderungen am liebsten und es ist ihm nicht wichtig, wie gut er ist.
Spieler mit Ehrgeiz	Dieser Typ von Spieler hat das Ziel so gut wie nur irgend möglich zu sein. Dabei ist es für ihn zweitrangig, wie viel Zeit er dafür aufwendet. Er möchte sich mit anderen Spielern vergleichen können.

Problemstellung

Um ein erfolgreiches Spiel zu entwickeln, müssen mögliche Varianten und Abläufe gefunden und evaluiert werden. Dieser Prozess stellt den Kern der eigentlichen Spielentwicklung dar. In diesem Kapitel werden Spielabläufe und Designvarianten untersucht. Weiter werden Entscheidungen getroffen, die den Rahmen der Umsetzung vorgeben.

Eingabe des Lösungswortes

Es gibt verschiedene Möglichkeiten, die Lösung einzugeben. Beim iPhone ist der Bildschirm die einzige Eingabefläche.

Der Ausgangspunkt ist, dass ein zu lösendes Anagramm auf dem Bildschirm dargestellt wird. Es bestehen die folgenden Möglichkeiten der Eingabe.

- Eine Tastatur wird angezeigt, die alle Buchstaben des Alphabets enthält und ein Textfeld, wo man die Lösung eingeben kann.
- Eine Tastatur wird angezeigt, bei der nur die Buchstaben auswählbar sind, die in der Lösung vorkommen und noch nicht eingegeben wurden. Die Lösung wird dabei auch in ein Textfeld eingegeben.
- Die Buchstaben des Anagramms werden als verschiebbare Elemente auf dem Bildschirm dargestellt und der Spieler muss sie durch verschieben mit dem Finger in die richtige Reihenfolge bringen.
- Unter dem Anagramm werden Lösungsboxen für jeden Buchstaben angezeigt. Beim Tippen auf eine solche Lösungsbox wird eine Tastatur angezeigt, mit deren Hilfe man den Lösungsbuchstaben für dieses Feld eingeben kann.

Es ist festzustellen, dass der Einsatz einer Tastatur den Spieler auf eine bisher nicht beschriebene Art fordert. Während des Überlegungsprozesses hat der Spieler nur das Anagramm im Blick und sucht den ersten bzw. nächsten Buchstaben, der der Lösung entspricht. Im folgenden muss der Spieler den Buchstaben, den er als potentiell richtigen im Sinn hat, auf der Tastatur suchen und eingeben. Dieser weitere Suchprozess vermindert bei ungeübten Spielern auf Dauer die Konzentration und dies trübt den Spaß, der beim Spielen entsteht.

Die Variante mit den frei verschiebbaren Buchstaben hat dieses Problem nicht und stellt die Variante dar, die bei der Realisierung von iDictAttack das Mittel der Wahl ist.

Hilfe

Für unerfahrende Spieler oder Nutzer, die iDictAttack das erste mal starten, kann es schwierig sein, sich zurecht zu finden. Hierfür muss eine Funktion im Spiel vorgesehen sein, die den Nutzer bei Bedarf unterstützen kann.

- Es besteht die Möglichkeit auf der Benutzeroberfläche permanent ein Hilfesymbol anzuzeigen. Bei der Auswahl dieses Symbols wird eine kontextbezogene Hilfe angezeigt.
Vorteil: In jeder Situation steht eine Hilfe zur Verfügung.
Nachteil: Sobald man erste Erfahrungen mit dem Spiel gesammelt hat, benötigt man die Hilfefunktion praktisch nicht mehr. Folglich kann das Hilfesymbol auch stören.
- Es ist möglich an einer gut erreichbaren Stelle im Programm eine Hilfefunktion bereitzustellen. Diese umfasst zu allen Hauptfunktionen eine detaillierte Beschreibung.
Vorteil: Es ist klar, wo die Hilfefunktion zu finden ist. Man wird nicht mit "Hilfsangeboten" belästigt.
Nachteil: Ein Erstnutzer kann sich unter Umständen allein gelassen fühlen.

Eine zentrale Hilfefunktion, in der alle Hauptfunktionen erläutert werden, belästigt geübte Nutzer nicht und ist einfach umzusetzen. iDictAttack wird eine solche Hilfefunktion anbieten.

Vergleichsmöglichkeit von Erfolgen

Besonders *Spieler mit Ehrgeiz* brauchen für Gewöhnlich eine Möglichkeit, ihre erreichten Erfolge untereinander zu vergleichen.

Aus den Entwicklern bereits bekannten Spielen sind die folgenden Möglichkeiten bekannt.

- Endpunktzahl: Es wird am Ende des Spiels eine Zahl angezeigt, die im Verhältnis zur erzielten Leistung steht. Diese wird gespeichert und kann im Hauptmenü abgerufen werden. Es wird nicht gespeichert, wer diese Punktzahl erspielt hat. Das Mobiltelefon Nokia 3210 [NOKI] hat sich beim Spiel Snake [SNAK] so verhalten.
- Bestenliste: Wie bei der Variante Endpunktzahl wird am Ende des Spiels eine Punktzahl angezeigt. Hier kann nun jedoch auch eine Name eingegeben werden. Über das Hauptmenü kann nun eine Liste aufgerufen werden, in welcher alle bisher erreichten Punktestände mit den dazugehörigen Namen verzeichnet ist. Sehr viele heutige Spiele haben diese Art Bestenliste.
- Synchronisierte Bestenliste: Diese Variante baut auf der soeben beschriebenen Bestenliste auf. Hier werden zusätzlich die erreichten Punktzahlen und Namen mit einem im Internet befindlichen Server abgeglichen. Jeder Nutzer des Spiels kann über das Hauptmenü eine globale Bestenliste aufrufen, wo man alle weltweit erreichten Punktestände und Namen betrachten kann. Dieses Verhalten ist bei allen heutigen Spielekonsolen verbreitet.

Für die Realisierung von iDictAttack wird auf eine konventionelle Bestenliste zurückgegriffen, wobei jeder Schwierigkeitsgrad in einer eigenen Liste geführt wird.

Schwierigkeitsgrade

Um verschiedenen Nutzergruppen das Spielen von iDictAttack zu ermöglichen, müssen verschiedene Schwierigkeitsgrade angeboten werden.

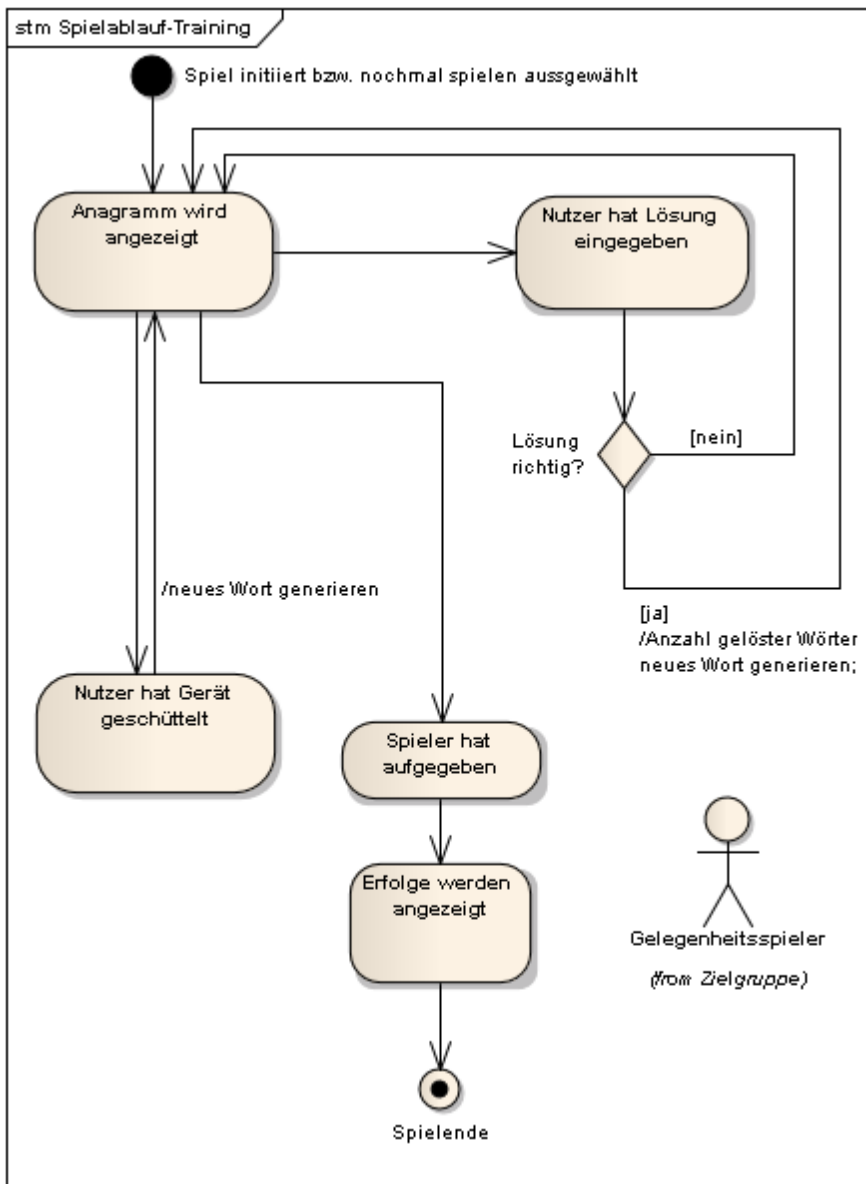
In den folgenden Kapiteln werden die Schwierigkeitsgrade und Spielerbewertungskriterien beschrieben, die in einem Treffen der Entwickler ausgearbeitet wurden.

Trainingsmodus

Der Trainingsmodus zeichnet sich dadurch aus, dass er den Spieler nicht unter Druck setzt. Ein Spieler kann ohne Einschränkungen ein Anagramm nach dem anderen lösen. Er wird nicht bestraft, wenn er ein Anagramm ohne es zu lösen überspringt. Bei diesem Modus wird in der Bestenliste die Anzahl der gelösten Wörter einer Spielsitzung geführt.

Dieser Schwierigkeitsgrad ist besonders für *Gelegenheitsspieler* gedacht, die sich unkompliziert die Zeit vertreiben möchten.

Das folgende Diagramm illustriert den Spielablauf dieses Schwierigkeitsgrads näher.



3: Spielablauf-Training

Im Diagramm ist zu sehen, dass ein Nutzer jederzeit mittels Schütteln ein neues Wort bekommen kann, wenn er das Gerät schüttelt. Sobald eine richtige Lösung eingegeben wurde, wird ein neues Anagramm angezeigt. Wenn ein Spieler aufgegeben hat oder keine Leben mehr besitzt, werden seine Erfolge angezeigt, wo er seinen Punktestand einsehen kann und diesem einen Namen zuordnen kann.

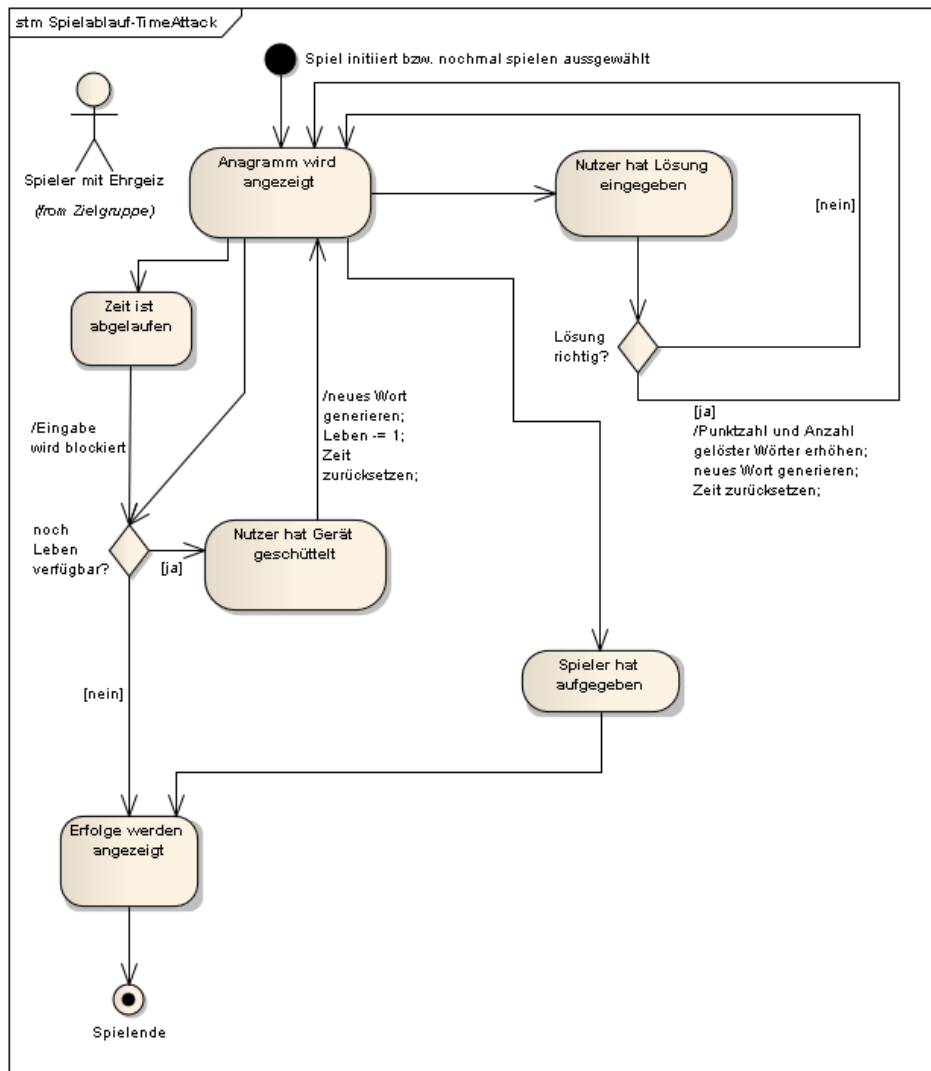
Diagrammelement	Beschreibung
Anagramm wird angezeigt	Es wird dem Spieler ein Anagramm angezeigt, dessen einzelne Buchstaben mit dem Finger verschoben werden können.
Nutzer hat Lösung eingegeben	Der Nutzer hat einen Buchstaben mit dem Finger bewegt. Das Spiel prüft, ob der Spieler die korrekte Lösung des Anagramms gefunden hat.
Nutzer hat Gerät geschüttelt	Der Nutzer hat das iPhone geschüttelt. Das Spiel gibt ihm ein neues Anagramm.
Spieler hat aufgegeben	Der Spieler hat die Spielsitzung aufgegeben.
Erfolge werden angezeigt	Dem Spieler werden seine erzielten Erfolge angezeigt. Weiter kann er seinen Namen eingeben, um dem von ihm erreichten Erfolg in der Bestenliste einen Namen zuzuordnen.
Lösung richtig?	Es wird geprüft, ob die Lösung des Anagramms korrekt ist.
Spiel initiiert bzw. nochmal spielen aussgewählt	Es wurden alle Spielzustände des letzten Spiel, wie gelöste Wörter, zurückgesetzt.
Spielende	Die Spielsitzung ist nun beendet.

Gegen die Zeit

Der Spielmodus *Gegen die Zeit* richtet sich primär an Fortgeschrittene Spieler. Dieser Modus fordert den Spieler, indem es ihm nur eine begrenzte Zeit für die Findung der Lösung zur Verfügung stellt. Weiter hat er nur eine begrenzte Anzahl an Leben zur Verfügung. In der Bestenliste werden die im Spiel erreichten Punktzahlen geführt.

Dieser Schwierigkeitsgrad ist besonders für *Spieler mit Ehrgeiz* gedacht, da sie hier von iDictAttack aktiv unter Druck gesetzt werden.

Das folgende Diagramm illustriert den Spielablauf dieses Schwierigkeitsgrads näher.



4: Spielablauf-TimeAttack

Im Diagramm ist zu sehen, dass ein Nutzer jederzeit durch Schütteln ein neues Anagramm bekommen kann. Dabei wird aber ein Leben abgezogen. Weiter existiert ein Timer. Sobald dieser abgelaufen ist, kann keine Eingabe mehr erfolgen. Sobald ein Anagramm richtig gelöst wurde, wird die Punktzahl und die Anzahl gelöster Wörter erhöht und ein neues Anagramm generiert. Ein Spieler kann jederzeit aufgeben.

Diagrammelement	Beschreibung
Anagramm wird angezeigt	Es wird dem Spieler ein Anagramm angezeigt, dessen einzelne Buchstaben mit dem Finger verschoben werden können.
Nutzer hat Lösung eingegeben	Der Nutzer hat einen Buchstaben mit dem Finger bewegt.

Zeit ist abgelaufen	Die verbleibende Bearbeitungszeit für das aktuell angezeigt Anagramm ist abgelaufen.
Nutzer hat Gerät geschüttelt	Der Nutzer hat das iPhone geschüttelt. Das Spiel gibt ihm ein neues Anagramm und verringert die Anzahl der Leben um eins.
Spieler hat aufgegeben	Der Spieler hat die Spielsitzung aufgegeben.
Erfolge werden angezeigt	<p>Dem Spieler werden seine erzielten Erfolge angezeigt. Weiter kann er seinen Namen eingeben, um dem von ihm erreichten Punktestand in der Bestenliste einen Namen zuzuordnen.</p> <p>Der Punktestand errechnet sich durch die folgende Formel.</p> $\text{Punkte} = \text{gelöste Wörter} \times 100 + \text{Anzahl Leben} \times 10$ <p>Es ist zu sehen, dass nur das Lösen der Anagramme zu hohen Punktzahlen führt. Die verbleibende Anzahl an Leben ist in die Formel eingeflossen, um Spielern, die evtl. aus Zeitgründen vorzeitig aufgeben, einen kleinen Bonus zu geben. Weiter werden durch diese Maßnahme frustrierte Spieler unter Umständen ein wenig aufgebaut.</p>
Lösung richtig?	Es wird geprüft, ob die Lösung des Anagramms korrekt ist.
noch Leben verfügbar?	Es wird geprüft, ob noch Leben verfügbar sind.
Spiel initiiert bzw. nochmal spielen ausgewählt	Es wurden alle Spielzustände des letzten Spiel, wie gelöste Wörter, Punktzahl, Zeit, Leben usw., zurückgesetzt.
Spielende	Die Spielsitzung ist nun beendet.

Benutzeroberflächen

Die Entwickler haben sich aus den folgenden Gründen entschieden, iDictAttack als NavigationBased-Application zu realisieren.

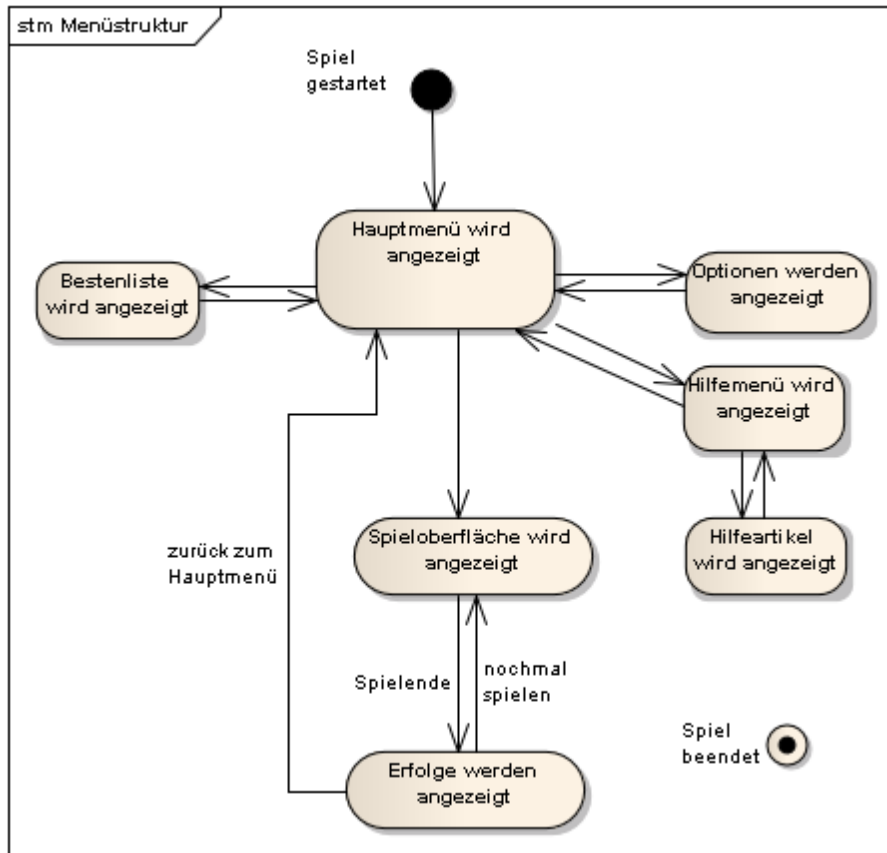
- Die Benutzeroberfläche wird mit mehreren Views ausgestattet sein und diese müssen über eine klar ersichtliche und einfach bedienbare Menüstruktur miteinander verknüpft werden.
- Obwohl eine NavigationBased-Application für Spiele optisch nicht gerade eine außergewöhnliche Form besitzt, ist jeder Nutzer mit deren Bedienung vertraut.
- Die Nutzung bereits fertiger Komponenten verkürzt die Entwicklungszeit.

Alle Views müssen sowohl im Landscape- als auch im Portraitformat des iPhones funktionieren.

In diesem Kaptiel werden alle Benutzeroberflächen grob beschrieben und deren Abhängigkeiten und Besonderheiten aufgezeigt.

Menüstruktur

Im folgenden Diagramm werden die Abhängigkeiten der verschiedenen Views illustriert. Die Abläufe innerhalb von *Spieloberfläche wird angezeigt* wurden im Kapitel *Schwierigkeitsgrad* näher erläutert.



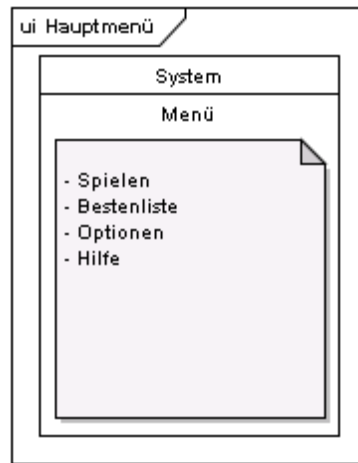
5:Menüstruktur

Diagrammelement	Beschreibung
Hauptmenü wird angezeigt	Das Hauptmenü erscheint als erstes nach dem Spielstart und stellt den Hauptknotenpunkt der Navigation dar.
Bestenliste wird angezeigt	Hier werden alle erreichten Punktzahlen in einer Tabelle angezeigt.
Optionen werden angezeigt	Hier wird ausschließlich ein Umschalter für den Schwierigkeitsgrad angezeigt. In einer nächsten Version könnte hier ebenfalls die Auswahl der Sprache, das Zurücksetzen der bereits gespielten

	Wörter, das Zurücksetzen der Bestenliste oder ähnliches erfolgen.
Hilfemenü wird angezeigt	In diesem Menü werden alle vorhandenen Hilfeartikel angezeigt. Die Oberfläche gleicht der des Hauptmenüs. lediglich im Menü des Navigation existiert ein Knopf, mit dem man zurück ins Hauptmenü gelangt.
Hilfeartikel wird angezeigt	Dieser View besteht aus einem Webbrowserfenster, welches den im Hilfemenü ausgewählten Hilfeartikel auf dem gesamten Bildschirm darstellt.
Spieloberfläche wird angezeigt	In diesem View wird die eigentliche Spieloberfläche dargestellt. Dies umfasst die einzelnen Buchstaben des aktuell zu spielenden Anagramms und eine Box mit aktuellen Spielinformationen.
Erfolge werden angezeigt	Es werden dem Nutzer seine erreichten Erfolge angezeigt.
Spiel gestartet	Nach dem Spielstart wird unmittelbar das Hauptmenü angezeigt.
Spiel beendet	Das Spiel kann jederzeit beendet werden.

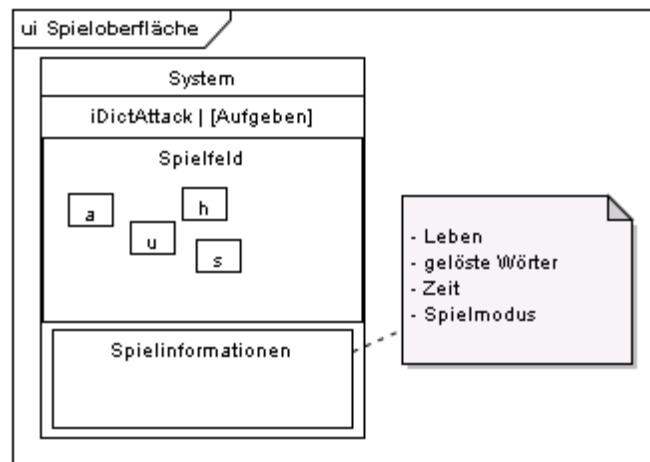
Entwürfe

Auf Basis der soeben definierten Menüstruktur wurden die folgenden Oberflächenentwürfe angefertigt.



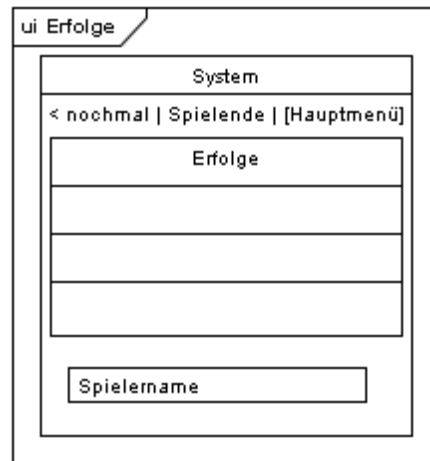
6:Hauptmenü

Wenn man *Spielen* auswählt, gelangt man in ein neues Spiel mit dem unter Optionen eingestellten Schwierigkeitsgrad.



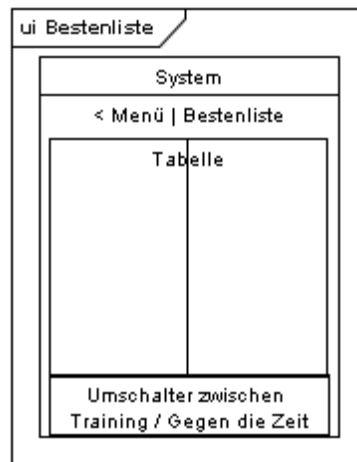
7:Spieloberfläche

Auf dem Spielfeld können die Buchstaben mit dem Finger verschoben werden. Mit dem Knopf "Aufgeben" in der Navigation wird das Ende des laufenden Spiels erzwungen.



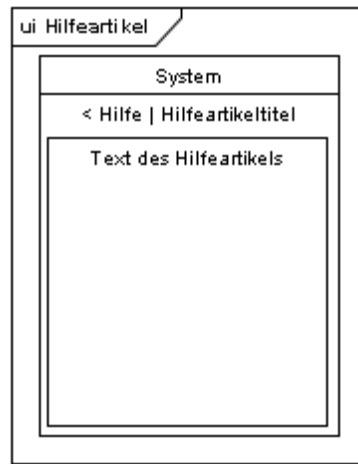
8:Erfolge

Im Menü der Navigation kann man entweder direkt nochmal spielen oder zum Hauptmenü zurückkehren. Durch drücken eines dieser Knöpfe wird der im Textfeld eingetragene Spielername und dessen Erfolge in der Bestenliste gespeichert.



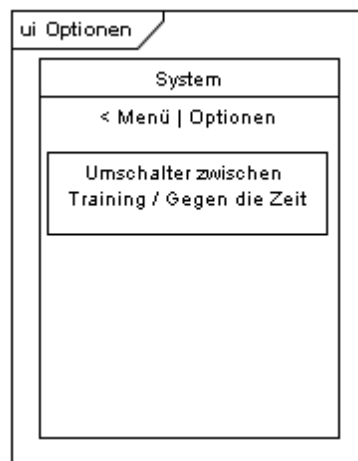
9:Bestenliste

Für jeden Schwierigkeitsgrad ist eine eigene Tabelle vorgesehen. Die beiden Tabellen werden mit einem Umschalter unter der Tabelle gewechselt.



10:Hilfeartikel

Durch Verlinkungen kann auf externe Webseiten zugegriffen werden. Zum Hilfemenü gelangt man durch den Knopf in der Navigation.



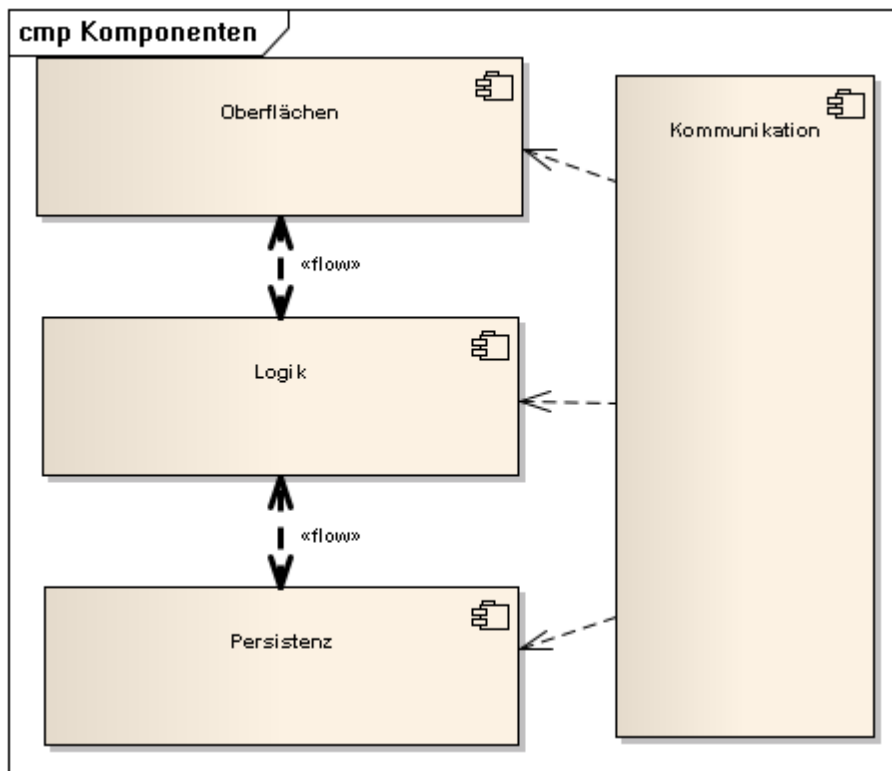
11:Optionen

Beim Betätigen des Umschalters wird die neue Einstellung sofort gespeichert.

Technische Konzepte

In diesem Kaptiel werden alle technischen Konzepte von iDictAttack dargestellt. Dies umfasst mehrere Komponenten und Klassendiagramme. Sämtliche hier dargestellten Umsetzungen fußen auf den vorherigen Analysen.

Komponenten



12:Komponenten

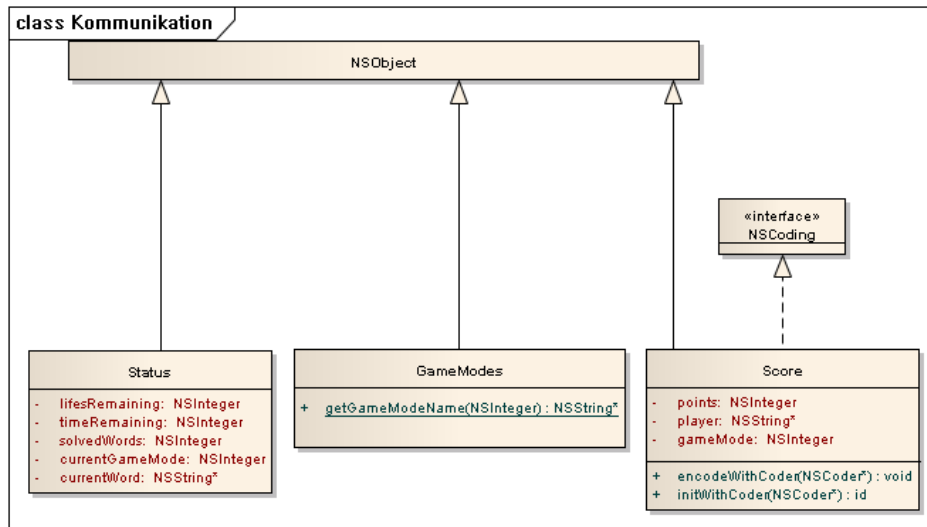
iDictAttack besteht rein prinzipiell aus drei Schichten.

- Die Oberflächen stellen die Schnittstelle zum Spieler dar. Ihre Aufgabe ist es, alle relevanten Inhalte, abhängig von der Situation darzustellen.
- Die Logikschicht umfasst alle relevanten Algorithmen und Abläufe, die iDictAttack ausmachen.
- In der Persistenzschicht werden alle relevanten Informationen gespeichert. Sie ermöglicht es, dass Informationen über verschiedene Programmmläufe hinweg bestehen bleiben und aktualisiert werden können.
- Die auf allen Schichten verfügbare Kommunikation stellt Datenstrukturen bereit, die in allen Schichten

zum Betrieb benötigt werden. Konkret werden in der Kommunikation Datentransfer-Objekte definiert, die in allen anderen Schichten verarbeitet werden.

Kommunikation

Im folgenden werden alle kommunikationsspezifischen Klassen illustriert und technisch beschrieben.

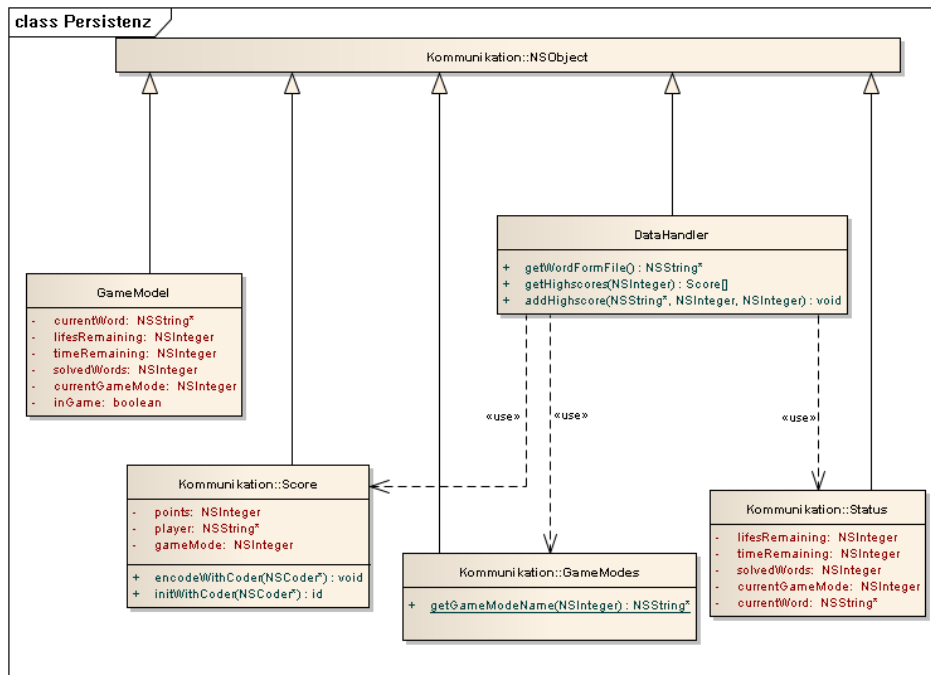


13:Kommunikation

Diagrammelement	Beschreibung
GameModes	<p>GameModes definiert alle verfügbaren Schwierigkeitsgrade.</p> <p>Konkret werden hier mithilfe von Defines NSInteger-Werte definiert, die in der gesamten Applikation eingesetzt werden können, um auf Schwierigkeitsgrade verweisen zu können.</p> <p>Es dient als Informationsquelle statischer Informationen.</p>
Score	<p>Score definiert, woraus ein Punktestand besteht. Es ist ein Datentransferobjekt. Sämtliche Attribute können über den Property-Mechanismus gelesen und beschrieben werden.</p>
Status	<p>Status definiert einen umfassenden Spielzustand. Es ist ein Datentransferobjekt. Sämtliche Attribute können über den Property-Mechanismus gelesen und beschrieben werden.</p>
NSObject	Standardobjekt der in Foundation
NSCoding	für API-eigene Serialisierung aus Foundation

Persistenz

Im folgenden werden alle persistenzspezifischen Klassen und Dateien illustriert und technisch beschrieben.

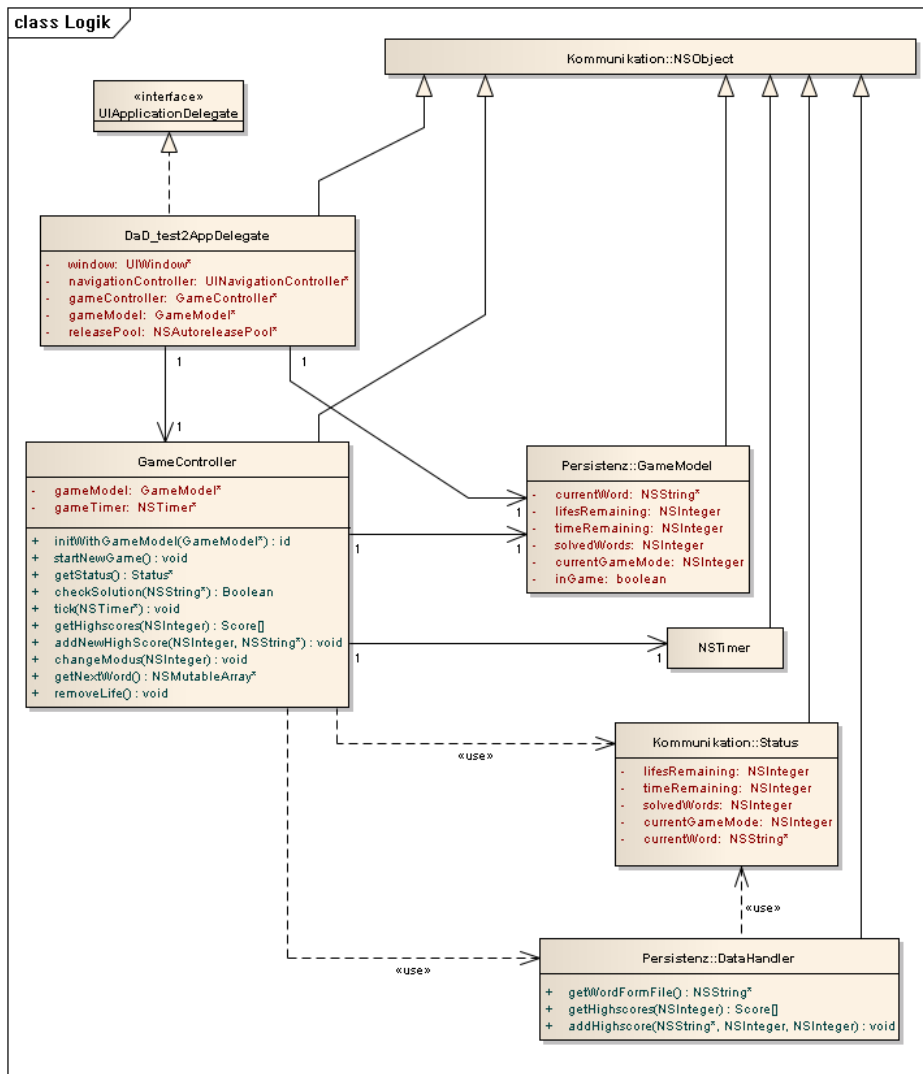


14:Persistenz

Diagrammelement	Beschreibung
DataHandler	<p>Führt den Dateizugriff durch.</p> <ul style="list-style-type: none"> Die initiale Wortliste liegt im AppBundle und wird beim ersten Start in <i>Documents/Words.txt</i> kopiert, um ab diesem Zeitpunkt bearbeitbar zu sein. Dort werden alle bereits gespielten Wörter abgestrichen. Wenn alle Wörter gespielt wurden, wird es wieder zurückgesetzt. Die Wortliste ist eine Textdatei, wo in jeder Zeile ein Wort steht. Ist ein Wort abgestrichen, beginnt es mit einer "#" Die Punktzahlen werden in einem serialisiertem Array in <i>Documents/highscores.dat</i> gespeichert.
GameModel	<p>Definiert, woraus der aktuelle Spielzustand besteht. Sämtliche Attribute können über den Property-</p>

Logik

Im folgenden werden alle logikspezifischen Klassen illustriert und technisch beschrieben.



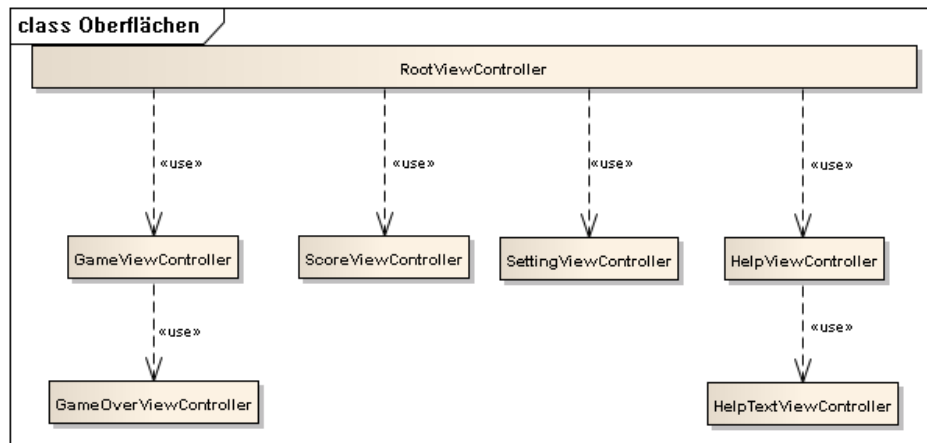
15:Logik

Diagrammelement	Beschreibung
DaD_test2AppDelegate	AppDelegate, woran die gesamte Applikation aufgehängt ist. Sämtliche Attribute können über den Property-Mechanismus gelesen und beschrieben werden.

GameController	Beinhaltet die gesamte Spiellogik; Sämtliche Attribute können über den Property-Mechanismus gelesen und beschrieben werden.
NSTimer	Ein Timer vom Foundation-Framework
UIApplicationDelegate	vom Framework UIKit

Oberflächen

Im folgenden werden alle Oberflächen samt ihrer Abhängigkeiten illustriert. Aus Platzgründen wird an dieser Stelle auf eine detaillierte Beschreibung verzichtet.



16:Oberflächen

Diagrammelement	Beschreibung
RootViewController	Stellt das eigentliche Hauptmenü dar;
GameViewController	Stellt das eigentliche Spielfeld dar;
GameOverViewController	Stellt die erreichten Erfolge nach einem Spielende dar;
HelpViewController	Stellt ein Menü über alle Hilfethemen dar;
HelpTextViewController	Stellt das eigentliche Hilfethema in einem Browser dar;
ScoreViewController	Stellt alle Bestenlisten des Spiels dar;
SettingViewController	Stellt auswählbare Optionen dar, die das Spiel beeinflussen;

Fazit

Alles in allem ist zu sagen, dass die Entwicklung selbst eines verhältnismäßig kleinen Computerspiels einen immensen Aufwand nach sich zieht. Alle am Projekt beteiligten Entwickler haben mit diesem Projekt und der damit verbundenen Entwicklung für ein Apple iDevice Neuland betreten.

Eine Veröffentlichung von iDictAttack im AppStore von Apple ist zum Zeitpunkt der Fertigstellung dieses Projektes nicht möglich. Apple verhindert die Veröffentlichung von quelloffener Software welche auf freie Lizenzen zurückgreift.

Weblinks

Auf die folgenden Webseiten wird in diesem Projektbericht verwiesen.

[TETR] Tetris - The Official Web Site, The Tetris Company, <http://www.tetris.com>

[SNAK] Snake - Wikipedia, die freie Enzyklopädie, verschiedene Autoren, OldID 84732374, <http://de.wikipedia.org/wiki/Snake>

[IPHO] iPhone - Mobiltelefon, iPod und Internetgerät, Apple Inc., <http://www.apple.com/de/iphone/>

[NOKI] Nokia Europe - Nokia 3210, Nokia Corp., <http://europe.nokia.com/support/product-support/nokia-3210>

...