



FACULTY OF TECHNICAL SCIENCES  
Institute of Information Technology

---

Christopher Robin Hribar  
*Angewandte Informatik*  
*Alpen-Adria-Universität Klagenfurt*  
Eberndorf, Österreich  
chhribar@edu.aau.at

# Gamification - Softwaresysteme spielerisch erlernen

---

BACHELORARBEIT

zur Erlangung des akademischen Grades  
Bachelor

Studium  
ANGEWANDTE INFORMATIK

Alpen-Adria-Universität Klagenfurt  
Fakultät für Technische Wissenschaften

Begutachter:  
Assoc.-Prof. Dr. Dipl.-Ing. Mathias Lux  
Institut für Informationstechnologie

September 2019

## **Ehrenwörtliche Erklärung**

Ich erkläre ehrenwörtlich, dass ich die vorliegende Schrift verfasst und die mit ihr unmittelbar verbundenen Arbeiten selbst durchgeführt habe. Die in der Schrift verwendete Literatur sowie das Ausmaß der mir im gesamten Arbeitsvorgang gewährten Unterstützung sind ausnahmslos angegeben. Die Schrift ist noch keiner anderen Prüfungsbehörde vorgelegt worden.

Klagenfurt, im September 2019

## **Danksagung**

An dieser Stelle möchte ich mich als Erstes bei meinem Bachelorbetreuer Mathias Lux für die stets gute Zusammenarbeit bedanken. Die Alpen-Adria-Universität ermöglichte es mir, mich weiterzubilden und stellte den Zugang zu Wissen bereit. Danke an diejenigen die mich bei diesem Thema unterstützt haben und vor allem ein großes Dankeschön an die Probanden meiner Studie, die sich freiwillig entschlossen haben bei dieser teilzunehmen.

## Zusammenfassung

Gamification ist der Gebrauch von Spielelementen in einem spiel-fremden Kontext [15]. Diese Methode gewinnt immer mehr an Interesse, nicht nur akademisch sondern auch industriell wird Gamification erforscht und eingesetzt. Es gibt einige Ergebnisse, wie spielerisches Lernen den Lernprozess beeinflussen [14], jedoch meist im pädagogischen Umfeld. Der Ansatz kann aber helfen neue Softwaresysteme spielerisch und selbstständig zu erlernen. Diese Arbeit beschäftigt sich mit Gamification, deren Fokus auf dem eigenständigen Erlernen von neuen Softwaresystemen gelegt ist. Dazu wird eine eigene, plattformunabhängige Game-Engine entwickelt und als Prototyp in ein Projekt-Management-Tool integriert. Es wird Allgemein darauf eingegangen was "Gamification" bedeutet und wie es derzeit eingesetzt wird. Um zu zeigen, dass durch hinzufügen von Spielelementen der Lernprozess beeinflusst wird, wurde eine Mixed-Method-Studie durchgeführt. In der Studie wurden 10 Probanden in 2 Gruppen aufgeteilt, eine Gruppe mit gamifizierter Software und eine Gruppe mit dem unveränderten Projekt-Management-Tool. Die Ergebnisse deuten auf einen leichten Effekt, dass mit Gamification schneller ein Programm erlernt werden kann.

# Contents

|          |                                    |           |
|----------|------------------------------------|-----------|
| <b>1</b> | <b>Einführung</b>                  | <b>3</b>  |
| 1.1      | Motivation . . . . .               | 4         |
| 1.2      | Related Work . . . . .             | 5         |
| <b>2</b> | <b>Implementierung</b>             | <b>6</b>  |
| 2.1      | Avatar . . . . .                   | 6         |
| 2.2      | Quest . . . . .                    | 6         |
| 2.3      | Levelsystem . . . . .              | 10        |
| 2.4      | Game-Loop . . . . .                | 11        |
| 2.5      | Game-Engine . . . . .              | 11        |
| <b>3</b> | <b>Integration in GanttProject</b> | <b>14</b> |
| 3.1      | Warum GanttProject . . . . .       | 14        |
| 3.2      | Integration . . . . .              | 14        |
| <b>4</b> | <b>Studie</b>                      | <b>17</b> |
| 4.1      | Studienaufbau . . . . .            | 17        |
| 4.2      | Studienablauf . . . . .            | 17        |
| 4.3      | Studienergebnisse . . . . .        | 19        |
| <b>5</b> | <b>Schlussfolgerung</b>            | <b>21</b> |

# List of Figures

|     |   |    |
|-----|---|----|
| 1.1 | Abbildung aus Hamari, Does Gamification Work? [9] . . . . . | 4  |
| 3.1 | Erklärung des Questsystems. . . . .                         | 15 |
| 3.2 | Pop-Up für Erreichen des maximalen Levels. . . . .          | 16 |
| 4.1 | Fragen zur Gamification. . . . .                            | 19 |

# List of Tables

4.1 Zeit in Sekunden der Testpersonen . . . . . 19



Der Ansatz von Gamification ist es, Computerspielelemente in einen spiel-fremden Kontext einzuführen. Mittels Fortschritt ebenso wie mit Erfahrungspunkten soll eine Motivationssteigerung und dadurch eine Verbesserung des Lernerfolges erzielt werden. Damit ist es nicht mehr notwendig, große Einschulungen zu halten, weil die Benutzer selbst das System spielerisch erlernen können. Es werden künstliche Aufgaben und Belohnungen konstruiert, die ähnliche Effekte zu dem realen Pendant erzeugen.

Doch was macht die Faszination von Computerspielen aus? Wie Tynan Sylvester in seinem Buch "Designing Games" in dem Untertitel beschreibt "A guide to Engineering Experiences"[1], geht es für einen Spieleentwickler darum Erfahrungen zu generieren und Emotionen in dem Spieler/der Spielerin auszulösen. Kurz gesagt sind Computerspiele ein Medium der Unterhaltung. Vor allem bei Multiplayer-Spielen, also in Spielen in der mehr als eine Person zusammen oder gegeneinander spielen, ist auch der kompetitive Wettstreit von großer Bedeutung. Das wird besonders in der E-Sport-Szene bemerkbar, da mittlerweile Turniere mit Preisgeldern von bis zu \$34,292,599.00 [8] ausgetragen werden, indem Teams gegeneinander antreten.

Gamification gewann in den letzten Jahren immer mehr an Interesse, sowohl im industriellen wie auch im akademischen Bereich [7]. Ein Beispiel für den Anstieg im akademischen Bereich zeigt Figure 1.1 aus der Arbeit "Does Gamification Work?" [9]. Zu sehen sind Suchergebnisse aus gängigen Suchmaschinen für wissenschaftliche Arbeiten. 2008 wurde das erste Mal "Gamification" in einem Blog-Eintrag erwähnt und 2010 wurde der Term auch industriell weit verbreitet [7]. Seit jehar ist eine Steigerung von Interesse zu beobachten.

Auch in der Industrie gibt es Beispiele für Gamification. Es existieren etablierte Unternehmen, die sich auf Gamification spezialisieren [11]. Aber schon kleine Elemente, wie öffentliche Auszeichnungen und Bestenlisten, die in Plattformen wie zum Beispiel bei StackOverflow [10] integriert werden, zählen als Gamification. Hier wird ein Augenmerk auf öffentliche (auch von anderen Benutzer/innen einsehbar) Errungenschaften gelegt. Andererseits gibt es "private" Errungenschaften, die nur dem Avatar (Spielfigur

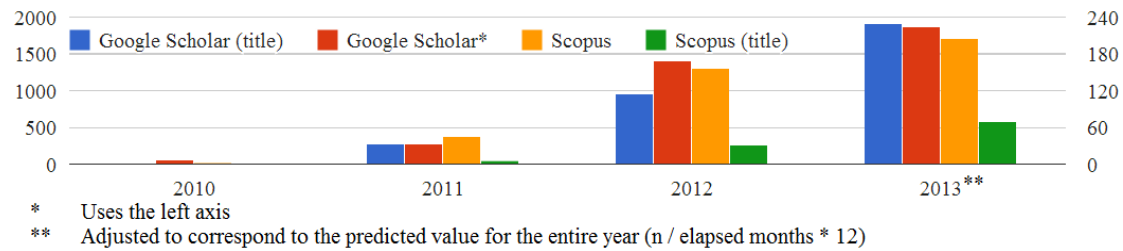


Figure 1. Search hits for "gamification"

Figure 1.1: Abbildung aus Hamari, Does Gamification Work? [9]

oder Spieler/in selbst) gutgeschrieben werden. So wird ein sogenannter "Skill-Tree" aufgebaut, der die Fähigkeit des Avatars abbildet und somit den Lernfortschritt widerspiegelt. Dies wird auch in dem Prototyp der Arbeit implementiert.

Ziel des Prototypen ist es, eine Game-Engine zu entwerfen die in möglichst beliebigen Softwareapplikationen integrierbar ist. Die erste Implementierung und Tests werden auf Basis einer Applikation angesetzt. Die Game-Engine beinhaltet ein Quest-System, das die Aufgaben modelliert und ein Levelsystem, das das Belohnungssystem widerspiegelt. Für die Studie wird der Prototyp in ein bestehendes Projekt-Management-Tool "GanttProject"[2] integriert.

Zum Abschluss wird mittels der Studie beobachtet, ob eine Tendenz zur Steigerung des Lernerfolges mit Gamification zu erkennen ist.

## 1.1 Motivation

Täglich werden über 1000 neue Apps in bekannten App-Stores veröffentlicht. Im Juni 2019 stehen alleine im Google Play Store 2,7 Millionen Apps zur Verfügung [3]. Dies alleine zeigt schon, dass es schwer ist, den Überblick zu behalten. Jedes Stück Software hat seine eigene Bedienung, doch ist es meist schwer den Funktionsumfang überschaubar zu halten. Nun geschieht es, zum Beispiel, dass neue firmeninterne Tools ausgerollt werden und mittels Einschulungen der Mitarbeiter das neue System integriert wird. Dies bedeutet aber hohe Einschulungs- und Supportkosten, Zeitaufwand und damit noch weitere Ausgaben für ein Unternehmen. Nicht nur im Unternehmerkontext kann ein neues System zu erlernen frustrierend und langwierig sein. Spielerisches Lernen zeigt besonders im Kindesalter Wirksamkeit und kann auch im Erwachsenenalter helfen, neue Systeme leichter zu lernen. Dies sind gute Voraussetzungen dafür, dass Spielelemente leichteres Lernen hervorrufen. Computerspiele sind, vereinfacht gesagt, Software mit einer zeitlichen Komponente und zusätzliche Aufgaben. Das bedeutet, dass jedes Stück Software mit Computerspielelementen ausgestattet und mit nur kleinen Änderungen

Spielelemente eingeführt werden können, um dadurch einzigartige Spiele zu entwickeln. Gamification kann auch für eine Marketing-Strategie herangezogen werden, ein Beispiel dafür ist ein Spiel welches für eine Fastfood-Kette von "gamewheel" entwickelt wurde. *"For the latest product promotion „The Angriest Whopper“ Burger King and their advertising agency setup München were looking for a new way to engage with their audience and get some additional traffic to their restaurants. The solution was a fast and casual game built on our innovative Game Ad Platform and integrated into Burger King's existing mobile app. The principle of the game is simple yet addictive: the player is challenged to collect 20 Jalapeños in under 20 seconds in order to unlock an exclusive discount coupon redeemable at the Burger King restaurants."* [12] Das Spiel wurde nach 14 Tagen 336,700 mal gespielt, es wurden dabei 7,866,146 Jalapeños gesammelt und 135,883 Gutscheine gewonnen.

Es existieren auch Studien, die zeigen, dass Gamification einen positiven Effekt auf die Motivation ausübt [9]. Die Probanden aller Studien gaben an, dass Gamification positive Erfahrungen bringen, doch ist "Freude" subjektiv und stark von dem Benutzer/der Benutzerin abhängig. Dieser Ansatz wird auch in dieser Arbeit weitergeführt und die Studie soll weitergehende Ergebnisse aufdecken.

## 1.2 Related Work

"Raising engagement in e-learning through gamification" [14] bietet eine Grundlage wie Gamification in der Bildung eingesetzt werden kann. Außerdem wird ein theoretisches Beispiel, wie eine "E-Learning"-Applikation gamifiziert werden kann, ausgearbeitet. Die Arbeit "Does Gamification Work? - A Literature Review of Empirical Studies on Gamification." [9] untersucht empirische Studien im Bereich von Gamification. Des Weiteren wird auf den derzeitigen Stand der Forschung eingegangen und zeigt Lücken in bereits existierender Literatur. Die Arbeit diente aufzuzeigen, dass ein akademisches Interesse an Gamification besteht und lieferte Beispiele von Studien. "Defining gamification: a service marketing perspective." [7] deckte zusätzlich auf, dass ein industrielles Interesse besteht und definiert Gamification für sich. Das Buch "Designing Games - A Guide to Engineering Experiences" stellt den Entwickler in Vordergrund und beschreibt Prinzipien und Designstrukturen die für die Spieleentwicklung förderlich sind. "Human-Computer Interaction" [6] beschreibt, unter anderem, Testmethoden im Bereich von Benutzerschnittstellen. Dies unterstütze den Aufbau der Studie, da Mensch-Computer Interaktion stark mit Computerspiele verknüpft ist.

Die Implementierung erfolgt in Java um eine größtmögliche Plattformunabhängigkeit zu erzielen. Für etablierte Computerspieler/innen ist die Terminologie in der Domain intuitiv. Spieleentwickler versuchen auch auf gewohnte und etablierte Terme und Methoden zurückzugreifen. Bedeutungen von einzelnen Termen variieren aber oft, deshalb werden nun die relevanten Spielelemente definiert und deren Implementation beschrieben. Einer der Hauptkomponente eines Spieles ist die "Game-Loop" (2.4). Wie der Name schon sagt, handelt es sich um eine endlose Schleife die durch Eingaben die "Game-Engine" (2.5) beeinflussen und den aktuellen Zustand ausgibt. Zusätzlich ist die vorhin genannte "Game-Engine" notwendig, da sie die allgemeine Spielmechanik abbildet. Die Spielmechanik setzt die Regeln, wie der Spielzustand manipuliert werden kann. Die Game-Engine beinhaltet in diesem Fall ein Questsystem (Aufgabenmodellierung) und ein Levelsystem (Erfolgsmodellierung). Des Weiteren wurde die gesamte Game-Engine mittels JUnit-Tests getestet und eine Code-Coverage von  $\geq 90\%$  erreicht.

## 2.1 Avatar

Ein Avatar ist eine künstliche Person oder ein grafischer Stellvertreter einer echten Person. Er kann also eine Spielfigur wie in "Mensch ärgere Dich nicht" sein, als auch ein Profil von unterschiedlichen Personen auf Internetplattformen. Im Fall des Prototypen ist der Avatar die Person, die die Applikation benützt.

## 2.2 Quest

Ein Quest ist eine Suche oder Betätigung, um etwas zu finden oder zu erhalten, kann aber auch in kleineren Aufgaben (Tasks) unterteilt werden. Ein Beispiel aus dem Alltag für ein Quest wäre "Einkauf" mit dem Unteraufgaben "kaufe Milch" und "kaufe Eier". Wenn als Erstes Milch gekauft wird, wurde eine Aufgabe zwar erledigt, jedoch der Quest

"Einkauf" noch nicht abgeschlossen. Erst beim erledigen der zweiten Aufgabe "kaufe Eier" sind nun alle Aufgaben der Quest erledigt und diese kann somit abgeschlossen werden. Bei den Abschlüssen von Quests werden Erfahrungspunkte gesammelt, die in das Levelsystem übergehen. Folgende Quests wurden für die Studie konstruiert:

**Quest 0:** "Willkommen bei Letzplay! - Viel Spaß beim spielen!"

Erforderliche Stufe: 0, Erfahrungspunkte: 10

**Task:** "Füge dich als Ressource hinzu! - Du bist der Projektplaner, also füge dich als Erstes hinzu!"

**Quest 1:** "Rollenerstellung - Füge nun alle Rollen hinzu"

Erforderliche Stufe: 1, Erfahrungspunkte: 5

**Task:** "Programmierung - Füge Programmierung den Rollen hinzu"

**Task:** "Geschäftsführung - Füge Geschäftsführung den Rollen hinzu"

**Task:** "Marketing - Füge Marketing den Rollen hinzu"

**Task:** "Finanzen - Füge Finanzen den Rollen hinzu"

**Quest 2:** "Erstelle geeignete Ressourcen - Du hast 3 Personen zur Verfügung: Chris ist der technisch Affine, Franz ist geschickt mit dem Umgang von Menschen und Julia behält den Blick für das große Ganze! Nun füge sie als Ressourcen hinzu"

Erforderliche Stufe: 1, Erfahrungspunkte: 5

**Task:** "Füge Chris hinzu - Füge Chris unter Ressourcen als Full-Stack-Developer hinzu"

**Task:** "Füge Franz hinzu - Füge Franz unter Ressourcen als Marketing hinzu"

**Task:** "Füge Julia hinzu - Füge Julia unter Ressourcen als Geschäftsführung hinzu"

**Quest 3:** "Milestones - Füge sie als Task hinzu und setze sie als Milestone"

Erforderliche Stufe: 2, Erfahrungspunkte: 5

**Task:** "Appentwicklung - Füge Appentwicklung als Task hinzu und markiere es als Milestone"

**Task:** "Marketingplanung - Füge Marketingplanung als Task hinzu und markiere es als Milestone"

**Task:** "Businessplan - Füge Businessplan als Task hinzu und markiere es als Milestone"

**Quest 4:** "Tasks - Füge nun die Aufgaben als Task hinzu"

Erforderliche Stufe: 2, Erfahrungspunkte: 50:

**Task:** "Backend - Füge die Aufgabe Backend hinzu"

**Task:** "Frontend - Füge die Aufgabe Frontend hinzu"

**Task:** "Infrastruktur - Füge die Aufgabe Infrastruktur hinzu"

**Task:** "Demo-Vorstellung - Füge die Aufgabe Demo-Vorstellung hinzu"

**Task:** "Werbekampagne - Füge die Aufgabe Werbekampagne hinzu"

**Task:** "Revenue-Model - Füge die Aufgabe Revenue-Model hinzu"

**Task:** "Umsatzplanung - Füge die Aufgabe Umsatzplanung hinzu"

**Quest 5:** "Dependencies - Füge nun alle Abhängigkeiten hinzu"

Erforderliche Stufe: 2, Erfahrungspunkte: 7

**Task:** "Infrastruktur->Backend - Backend ist von Infrastruktur abhängig"

**Task:** "Backend->Frontend - Frontend ist von Backend abhängig"

**Task:** "Frontend->Demo - Demo-Vorstellung ist von Frontend abhängig",

**Task:** "Appentwicklung->Werbekampagne - Die Appentwicklung muss vollständig sein, bevor die Werbekampagne starten kann"

**Task:** "Demo-Vorstellung->Werbekampagne - Das Produkt muss vorher vorgestellt werden, bevor die Werbekampagne starten kann"

**Task:** "Revenue-Model->Umsatzplan - Es muss erst klar sein wie Geld eingenommen werden kann, bevor der Jahresumsatz geplant werden kann"

**Quest 6:** "Ressourcenzuteilung - Du hast Ressourcen zur Verfügung und Aufgaben die erledigt werden müssen, teile nun die Ressourcen ein"

Erforderliche Stufe: 3, Erfahrungspunkte: 5

**Task:** "Backend:Programmierung - Teile die Aufgabe Backend der Programmierung zu"

**Task:** "Frontend:Programmierung - Teile die Aufgabe Frontend der Programmierung zu"

**Task:** "Infrastruktur:Programmierung - Teile die Aufgabe Infrastruktur der Programmierung zu"

**Task:** "Demo-Vorstellung:Marketing - Teile die Aufgabe Demo-Vorstellung den Marketing zu"

**Task:** "Werbekampagne:Marketing - Teile die Aufgabe Werbekampagne den Marketing zu"

**Task:** "Revenue-Model:Geschäftsführung - Teile die Aufgabe Revenue-Model den Geschäftsführer zu"

**Task:** "Umsatzplanung:Finanzen - Teile die Aufgabe Umsatzplanung den Finanzen zu"

**Quest 7:** "Zeiteinteilung - Teile nun allen Tasks die Zeit zu"

Erforderliche Stufe: 4, Erfahrungspunkte: 8

**Task:** "Backend: 75 Tage - Backend braucht 75 Tage"

**Task:** "Frontend: 84 Tage - Frontend braucht 84"

**Task:** "Infrastruktur: 60 Tage - Infrastruktur braucht 60 Tage"

**Task:** "Demo-Vorstellung: 5 Tage - Demo-Vorstellung braucht 5 Tage"

**Task:** "Werbekampagne: 38 - Werbekampagne braucht 38 Tage"

**Task:** "Revenue-Model: 131 - Revenue-Model braucht 131 Tage"

**Task:** "Umsatzplanung: 131 - Umsatzplanung braucht 131 Tage"

**Quest 5:** "Dependencies - Füge nun alle Abhängigkeiten hinzu"

Erforderliche Stufe: 2, Erfahrungspunkte: 7

**Task:** "Infrastruktur->Backend - Backend ist von Infrastruktur abhängig"

**Task:** "Backend->Frontend - Frontend ist von Backend abhängig"

**Task:** "Frontend->Demo - Demo-Vorstellung ist von Frontend abhängig",

**Task:** "Appentwicklung->Werbekampagne - Die Appentwicklung muss vollständig sein, bevor die Werbekampagne starten kann"

**Task:** "Demo-Vorstellung->Werbekampagne - Das Produkt muss vorher vorgestellt werden, bevor die Werbekampagne starten kann"

**Task:** "Revenue-Model->Umsatzplan - Es muss erst klar sein wie Geld eingenommen werden kann, bevor der Jahresumsatz geplant werden kann"

**Quest 6:** "Ressourcenzuteilung - Du hast Ressourcen zur Verfügung und Aufgaben die erledigt werden müssen, teile nun die Ressourcen ein"

Erforderliche Stufe: 3, Erfahrungspunkte: 5

**Task:** "Backend: Programmierung - Teile die Aufgabe Backend der Programmierung zu"

**Task:** "Frontend: Programmierung - Teile die Aufgabe Frontend der Programmierung zu"

**Task:** "Infrastruktur: Programmierung - Teile die Aufgabe Infrastruktur der Programmierung zu"

**Task:** "Demo-Vorstellung: Marketing - Teile die Aufgabe Demo-Vorstellung den Marketing zu"

**Task:** "Werbekampagne: Marketing - Teile die Aufgabe Werbekampagne den Marketing zu"

**Task:** "Revenue-Model: Geschäftsführung - Teile die Aufgabe Revenue-Model den Geschäftsführer zu"

**Task:** "Umsatzplanung: Finanzen - Teile die Aufgabe Umsatzplanung den Finanzen zu"

**Quest 7:** "Zeiteinteilung - Teile nun allen Tasks die Zeit zu"

Erforderliche Stufe: 4, Erfahrungspunkte: 8

**Task:** "Backend: 75 Tage - Backend braucht 75 Tage"

**Task:** "Frontend: 84 Tage - Frontend braucht 84"

**Task:** "Infrastruktur: 60 Tage - Infrastruktur braucht 60 Tage"

**Task:** "Demo-Vorstellung: 5 Tage - Demo-Vorstellung braucht 5 Tage"

**Task:** "Werbekampagne: 38 - Werbekampagne braucht 38 Tage"

**Task:** "Revenue-Model: 131 - Revenue-Model braucht 131 Tage"

**Task:** "Umsatzplanung: 131 - Umsatzplanung braucht 131 Tage"

## 2.3 Levelsystem

Erfahrungspunkte sind ein Maß für die Entwicklung des Avatars. Wenn also ein Quest abgeschlossen wurde, wird der Avatar mit Erfahrungspunkte (kurz: EP, EXP, XP) belohnt. Durch erledigen von Aufgaben sammelt der Avatar Erfahrungen. Meist erfolgt bei Erreichen einer bestimmten Menge von Erfahrungspunkte ein "Stufenaufstieg". Ein Stufe (auch Level) ist meist eine Zahl die mit 1 beginnt und bei einem Stufenaufstieg inkrementiert wird. Somit ist klar, dass ein Avatar mit Stufe 5 eine höhere Erfahrungsstufe hat als ein Avatar mit Stufe 1. Das implementierte Levelsystem startet bei Stufe 0 und einer kleinen Quest, nämlich sich selbst im Projekt hinzuzufügen. Dies dient als Einstieg, so gesehen ein kleines Tutorial, und nach der kleinen Aufgabe startet man mit Stufe 1 mit den davor beschriebenen Quests. Ziel ist es die Maximalstufe 5 zu Erreichen, dies geschieht nach Erledigen aller Quests und somit ist "das Spiel" beendet.



## 2.4 Game-Loop

Die Spielschleife ist eine endlose Schleife die solange besteht wie das Spiel läuft. Die Funktionen der Game-Loop sind die Verarbeitung des Inputs, die Aktualisierung des Spielzustandes und die Anzeige (Output), die regelmäßig wiederholt ausgeführt werden. Als Input dienen in der Implementierung Text-Trigger. Wird eine Funktion im Programm ausgeführt, übergibt diese einen Trigger als Input an den Prototypen. Beim Aktualisieren werden die Aufgaben erledigt, wenn sie durch einen richtigen Trigger im Input ausgelöst wurden. Die Ausgabe ist der Ist-Zustand des Spieles, also das derzeitige Level und verfügbare Aufgaben. Ein Auszug der GanttProject-GameLoop:

---

```
public class GanttGameLoop implements GameLoop {
    @Override
    public void init() {
        game.initQuests(GanttProject.init());
    }

    @Override
    public void input(String trigger) {
        game.trigger(trigger);
    }

    @Override
    public void update() {
        game.update();
    }

    @Override
    public Output output() {
        return new Output(game.availableQuests(), game.currentLvl(),
            game.currentExp());
    }
}
```

---

## 2.5 Game-Engine

So wie die Gesetze der Physik bestimmen, wie wir als Mensch mit unserer Umwelt physisch interagieren, beschreibt die Game-Engine, wie der Spieler/die Spielerin mit der Spielumgebung interagieren kann. In der Implementierung beinhaltet die Spielmechanik

ein Questsystem und ein Levelsystem.

Das Questsystem beinhaltet Aufgaben, die erledigt werden müssen, um die Quest abzuschließen. Bei Abschluss einer Quest werden Erfahrungspunkte gesammelt, welche in das Levelsystem übergehen. Aufgaben werden durch Trigger gelöst. So modelliert man zum Beispiel für die Aufgabe "Füge Chris als Ressource hinzu" den Trigger "res:add:chris". "res" steht für den Typ von einem Trigger, "add" die Funktion und "chris" ist der Wert. Kommt nun als Input der Trigger "res:add:chris" wird die vorher genannte Aufgabe gelöst und gegebenenfalls abgeschlossen. Ein Auszug aus der Questklasse:

---

```
public class Quest implements Trigger {
    private String title;
    private String description;
    private List<Task> taskList;
    private int minLvl;
    private float gainExp;
    private boolean finished;
    private List<String> triggers;

    public Progression progression() {
        int done = 0;
        for (Task task : taskList) {
            if (task.isDone()) {
                done++;
            }
        }
        return new Progression(done, taskList.size());
    }

    @Override
    public void trigger(String msg) {
        if (triggers.contains(msg)) {
            for (Task t : taskList) {
                t.trigger(msg);
            }
        }
    }
}
```

---

Das Levelsystem soll den Fortschritt des Lernerfolgs widerspiegeln. So sollen zum Beispiel nur "leichte" Quests bei niedrigem Level angezeigt werden und bei Erreichen einer höheren Stufe bekommt der Spieler dementsprechend "schwerere" Quests. Das Levelsystem beinhaltet die aktuelle Stufe und die gesammelten Erfahrungspunkte. Durch

---

erledigen einer Quest werden Erfahrungspunkte in Form von 1-100% gutgeschrieben. Sobald  $\geq 100\%$  erreicht wird, steigt der Benutzer eine Stufe auf. Die überschüssigen Punkte werden für den nächsten Stufenaufstieg gutgeschrieben.

Es ist auch möglich, den Spielstand zu speichern und auch zu laden, dies wurde für die Studie nicht benötigt.

Das Ziel ist es zu zeigen, dass Spielelemente in einer Software den Lernerfolg steigert. Der Prototyp wird in GanttProject integriert und die umgeänderte Software wird zur Studie herangezogen.

### 3.1 Warum GanttProject

GanttProject ist ein ausgereiftes Projekt-Management-Tool, welches 2003 gegründet wurde und ist als "Free and Open Source" sowie auch für kommerzielle Zwecke frei verwendbar. Es ist in Java geschrieben und dadurch läuft es auf jeder Plattform, auf der die Java Runtime verfügbar ist. [2]

Des Weiteren ist es ein unbekanntes Tool, das für die Studie hilfreich ist, da kein Vorwissen vorhanden sein darf. Aufgrund des wirtschaftlichen Nutzen ist ein Projekt-Plan-Tool zielführend, da es zeigt, dass für jede Software eine Gamification möglich ist. Das einzige Problem ist, dass Projekt-Management-Tools im Allgemeinen nicht unbekannt sind und dieses Vorwissen in der Studie auch erhoben und berücksichtigt wird.

### 3.2 Integration

Da der Sourcecode von GanttProject auf GitHub[4] online ist, wurde das Projekt auf ein privates Repository[5] verzweigt. Der Prototyp wird in einem eigenen Projekt entwickelt und als Package (.jar) exportiert, welches dann im GanttProject-Fork im Branch "letz-play" importiert wird. Es ist nun nur noch nötig die Trigger im Sourcecode auszulösen und die Ausgabe darzustellen. Beispielsweise wird unter hinzufügen einer Person im "HumanResourceManager", was die Personen-Ressourcen modelliert, der Trigger "res:add:NAME" hinzugefügt. Um den Rest kümmert sich der Prototyp. Wenn also ein Trigger den Zustand verändert wird dieser auch im GanttProject angezeigt. Die Anzeige,

bzw. Ausgabe der Game-Engine wird direkt in die Benutzeroberfläche integriert. Dazu wurde ein eigenes Panel im Programm hinzugefügt, das das aktuelle Level und die verfügbaren Quests anzeigt. Die Questansicht erfolgt über eine Baum-Darstellung und mittels anklicken wird die Beschreibung der ausgewählten Aufgabe darunter angezeigt, Abbildung: 3.1.

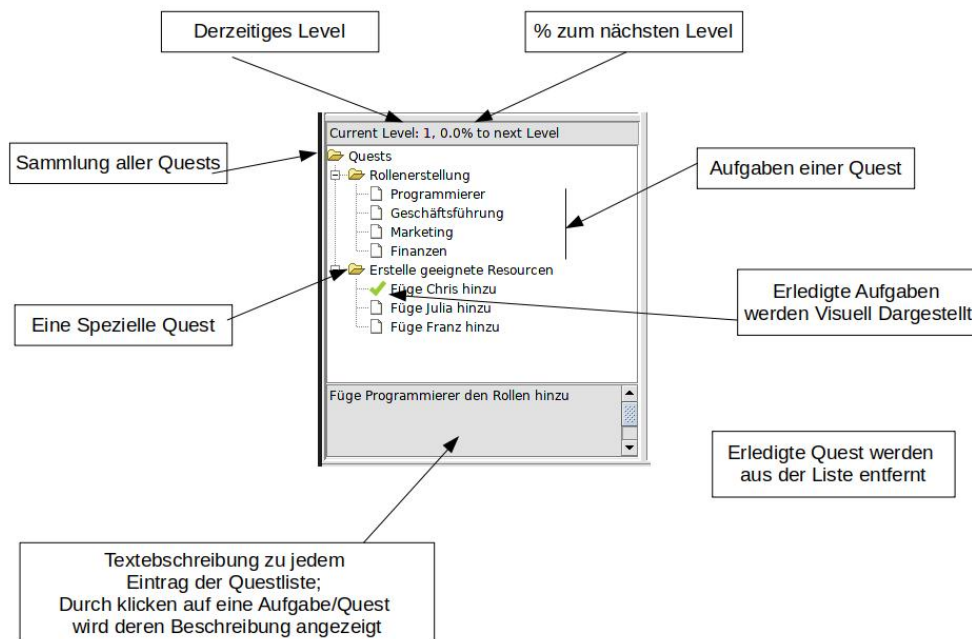


Figure 3.1: Erklärung des Questsystems.

Da das GanttProject aus vielen Modulen besteht, wird nun kurz darauf eingegangen wo welche Änderungen durchgeführt wurden. Nachfolgende Pfade werden relativ vom Root-Verzeichniss ".../ganttproject/" angegeben. Der Kern des Projektes ist unter dem Modul "ganttproject". Darin befinden sich die Library-Dateien, in der auch der Prototyp eingebunden wurde, sowie das "Source-Verzeichniss" (src). Unter /src/net/sourceforge/ganttproject befinden sich alle Modellierungen bezüglich "Tasks", "Resources" und "Roles". Diese drei genannten Objekte bilden die Grundlage für das Projekt-Management-Tool und sind für die Studie notwendig. Die Trigger für "Roles" wurde in der Klasse "RoleSetImpl" integriert und sieht folgend aus:

---

```
//creating role trigger
GanttGameLoop.getGameLoop().input(
    rol+add+name.toLowerCase().trim());
GameOutput.refresh();
```

---

Die Ressourcen-Trigger für Hinzufügen wurde in "HumanResourceManager" und für Ressourcenzuteilung in "ResourceAssignmentCollectionImpl" implementiert. Trigger für Tasks wurden in "TaskManagerImpl" hinzugefügt und die Abhängigkeiten in "TaskDependencyImpl".

Die Ausgabe (Abbildung: 3.1) wurde in einer eigenen Klasse entwickelt. Die Klasse "GameOutput" beinhaltet die "Swing"-Komponenten (Grafische Darstellung) und auch die Ausgabe von der Game-Engine, die in der Benutzeroberfläche von GanttProject angezeigt wird. Bei einem Levelaufstieg wird ein Pop-Up eingeblendet, Abbildung: 3.2.

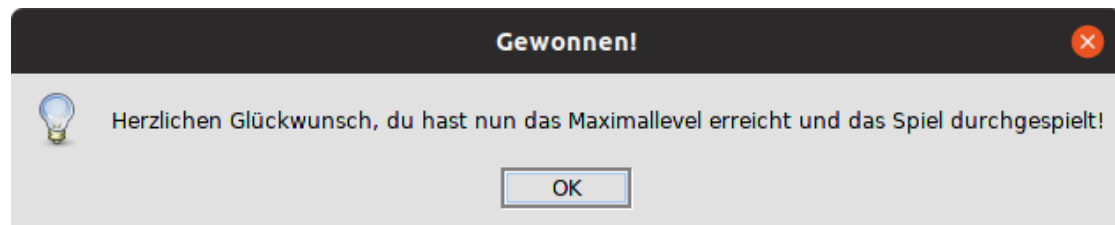


Figure 3.2: Pop-Up für Erreichen des maximalen Levels.

Um GanttProject zu kompilieren wird im Hauptverzeichnis der Befehl `./gradlew distbin` ausgeführt. Zum Starten begibt man sich unter `./ganttproject-builder/distbin` im Hauptverzeichnis und führt `./ganttproject` aus.

Die Studie soll eine mögliche Korrelation zwischen der benötigten Zeit zum Erledigen einer Aufgabe, in einer den Versuchspersonen unbekanntem Software, mit und ohne Gamification aufdecken. Die Hypothese ist, dass die Gruppe mit Gamification signifikant schneller ist als die Kontrollgruppe. Die Studie ist zu den Probanden komplett offen und transparent.

### 4.1 Studienaufbau

Es werden 10 Testpersonen ohne Vorwissen in GanttProject herangezogen. Die Probanden werden alternierend in zwei Gruppen aufgeteilt, das bedeutet, dass die erste Testperson der Gruppe A, die Zweite Gruppe B, die Dritte Gruppe A,... zugeteilt wurde. Die Kontrollgruppe (Gruppe A) bekommt die gleiche Aufgabe die sie nur mittels offizieller Dokumentation erledigen müssen, jedoch ohne die Integration der Game-Engine. Die Versuchsgruppe (Gruppe B) bekommt zusätzlich die gamifizierte Version der Software. Die Aufgaben sind identisch in beiden Gruppen und die Kontrollgruppe bekommt auch die Aufgaben portioniert wie im Questsystem. Der einzige Unterschied ist, dass die Versuchsgruppe B eine zusätzliche einführende Aufgabe, nämlich "Füge dich selbst als Ressource ein!" um das spielerische "Du bist Teil des Spieles" zu unterstützen. Die Messung erfolgt in Zeit, von dem Beginn bis zum erledigen der Aufgaben.

### 4.2 Studienablauf

Nachdem die Testperson der Gruppe zugewiesen wurden, wird zuerst der Studienaufbau beschrieben. Als nächstes wurde ein Fragebogen von dem Beobachter ausgefüllt. Der Fragebogen beschäftigt sich mit der Erhebung ob GanttProject bekannt bzw. verwendet worden ist. Letzteres führt zum Ausschluss der Studie, da der Lernerfolg nicht mehr beurteilt werden kann. Des Weiteren wird der bisherige Kontakt zu Computerspielen erhoben. Als Letztes wird die Muttersprache abgefragt und wie gut der Umgang mit

englischsprachiger Software ist. Da GanttProject und offizielle Dokumentation nur in englischer Sprache zur Verfügung steht, führen unzureichende Englischkenntnisse zum Ausschluss der Studie.

Als nächstes wird die offizielle Dokumentation, in diesem Fall ein Tutorial-Video, den Probanden bereitgestellt und sie werden aufgefordert, dieses anzusehen. Das offizielle Video [13] hat eine Länge von 15 Minuten und 4 Sekunden und zeigt umfassend den Funktionsumfang. Da viele Punkte besprochen werden, die nicht relevant für die Studie sind, wurde das Video auf 8 Minuten und 58 Sekunden zugeschnitten um die Testpersonen zu beschränken und nicht zu überfordern. Dieses Video kann jederzeit während der Studie abgerufen werden, jedoch nach Beginn der Zeitmessung fließt die Zeit für das Anschauen des Videos mit ein.

Nach dem Video bekommen die Versuchspersonen der Gruppe B (mit Gamification) die Beschreibung des Questsystems, Abbildung: 3.1.

Als letzter Schritt, bevor die Messung erfolgt, bekommen die Probanden das "Szenario" - sprich eine modellierte Aufgabenbeschreibung für eine Projektplanung die anschließend mit GanttProject umzusetzen ist. Das Szenario ist folgend beschrieben:

*Du hast eine Appidee für ein innovatives soziales Netzwerks. Du willst einen Förderantrag über €42.000 für die Umsetzung deines Projektes. Die Förderzeit beträgt 1 Jahr und dazu wird ein Projektplan für diesen Zeitraum benötigt. Des Weiteren stehen dir 3 Personen zur Verfügung. Erstelle nun ein Projektplan mit "GanttProject" mit folgenden Kriterien:[...]*

Weiters sind alle Aufgaben beschrieben, wie zum Beispiel, welche Ressourcen, Aufgaben und Rollen angelegt werden müssen. Insgesamt handelt es sich um 7 Aufgaben. Eine Liste der gesamten Aufgaben befindet sich im Kapitel "Implementation" unter "Quest" (2.2).

Gruppe B bekommt diese Aufgaben portioniert über das integrierte Questsystem und deswegen bekommt die Gruppe A die Aufgaben einzeln, um den gleichen Ablauf zu fixieren.

Nachdem alle Aufgaben erledigt wurden, wird die Zeitmessung beendet und als Letztes wird noch ein Fragebogen von dem Beobachter ausgefüllt. Die Erste Frage ist offen gestellt ("Wie war es?"[6]) damit die Versuchspersonen ohne eine Richtung vorzugeben antworten können. Danach wird noch eingeschätzt wie gut nun der Umgang mit GanttProject ist. Für die Gruppe B werden noch 4 Fragen nach der Likert-Skala bereitgestellt um abzufragen, ob die Gamifizierung geholfen hat. Siehe Abbildung: 4.1



**Zusätzliche Fragen mit LetzPlay**

**Mir haben die Spielelemente beim Erlernen von „GanttProject“ geholfen.**  
 trifft zu     trifft eher zu     trifft eher nicht zu     trifft nicht zu     Keine Angabe

**Mich haben die Spielelemente beim Erlernen von „GanttProject“ gestört.**  
 trifft zu     trifft eher zu     trifft eher nicht zu     trifft nicht zu     Keine Angabe

**Ich finde dass Gamification das Erlernen von Softwareapplikationen erleichtert.**  
 trifft zu     trifft eher zu     trifft eher nicht zu     trifft nicht zu     Keine Angabe

Figure 4.1: Fragen zur Gamification.

### 4.3 Studienergebnisse

Die Tabelle 4.1 zeigt die Rohdaten der Testpersonen in der jeweiligen Gruppe und deren Zeit zum Erledigen der Aufgabe.

Table 4.1: Zeit in Sekunden der Testpersonen

| Testperson | Gruppe | Zeit |
|------------|--------|------|
| 1          | A      | 1369 |
| 2          | B      | 980  |
| 3          | A      | 1050 |
| 4          | B      | 459  |
| 5          | A      | 809  |
| 6          | B      | 1140 |
| 7          | A      | 951  |
| 8          | B      | 655  |
| 9          | A      | 504  |
| 10         | B      | 866  |

Werden die zwei Gruppen verglichen und statistisch ausgewertet ist ein leichter Effekt zu beobachten. Die Gruppe B mit der Gamification ist im Schnitt schneller als die Kontrollgruppe A. Der Korrelationskoeffizient  $r$  beträgt  $-0,1023$  was den leichten Effekt

erklärt. Ob dieses Ergebnis aussagekräftig ist sei zu überprüfen, da nur 10 Testpersonen zur Studie herangezogen wurden. Ein Zusammenhang zwischen den Fragebogen und der Zeit ist nicht zu erkennen, sprich es war irrelevant ob die Probanden vorher schon in Kontakt mit Computerspielen getreten sind oder bereits Erfahrungen im Bereich der Projektplanung hatten. Während der Studie wurde beobachtet, dass Testpersonen die im Allgemeinen einen vertieften Umgang mit Computersystemen und Applikationen im Schnitt schneller waren, dies wurde jedoch nicht erhoben und ausgewertet. Das zu erheben ist ein Vorschlag im Fall einer Wiederholung der Studie.

Alle Testpersonen von der Gruppe B gaben an, dass die Spielelemente lustig, hilfreich und motivierend seien. Nur zwei der fünf Testperson der Gruppe A gaben an, dass das Erledigen der Aufgabe "spaß" gemacht hat. Zwei Testpersonen gaben an, dass die Mischung aus Deutsch und Englisch verwirrend sei, das ist weiters ein Verbesserungspunkt der Studie. Zwei Probanden der Gruppe B haben bei der Frage ob Gamification hilfreich sei mit "trifft eher zu" geantwortet, die restlichen 3 mit "trifft zu". Die Kontrollfrage, ob die Spielelemente gestört haben, gaben alle Personen "trifft nicht zu" an und nur eine Testperson von fünf haben bei der Frage "Ich finde dass Gamification das Erlernen von Softwareapplikationen erleichtert." mit "trifft eher zu" geantwortet, die anderen vier Personen gaben "trifft zu" an.

Bei zwei Aufgaben mussten die Probanden am häufigsten im Tutorialvideo nachsehen. Die Aufgaben sind zu einem, das Hinzufügen von Rollen. Diese Funktion ist nicht direkt in der Benutzeroberfläche in GanttProject integriert sondern muss über den Menüpunkt "Projekteigenschaften" hinzugefügt werden. Die zweite Aufgabe, die Abhängigkeiten der Tasks zu modellieren, hat die Mehrheit die Abhängigkeiten in falscher Reihenfolge aufgefasst. Ein Beispiel dafür ist *"Zuerst muss die „Infrastruktur“ bestehen um mit dem „Backend“ beginnen zu können"*. In GanttProject muss diese Abhängigkeit im Task "Backend" unter den "Predecessor", also den Vorgänger, der Task "Infrastruktur" angegeben werden. Dies jedoch wurde oft falsch verstanden. Der Beobachter gab bei falscher Zuweisung nur an, dass die Abhängigkeiten falsch zugewiesen wurde. Des Weiteren wurde beobachtet, dass die schnellste Testperson als einziges das Programm fast ausschließlich mit Tastatur und Tastaturkurzbefehle bediente, was die geringe Zeit erklärt.

Die Arbeit liefert Indizien, dass Gamification das selbständige Lernen unterstützen kann. Jedoch gibt es keine Garantie dafür, dass gamifizieren von Software immer den Lernprozess positiv beeinflusst. Kleine Computerspielelemente in beliebige Software zu integrieren ist dennoch möglich. Auch ein akademisches und industrielles Interesse an Gamification wurde aufgeklärt. Allgemein bleibt zu sagen, dass Computerspiele und gamifizierte Software noch am Anfang deren Entstehungsgeschichte befinden und immer weiter erforscht und eingesetzt werden. Um aussagekräftigere Ergebnisse zu erhalten, sollte die Studie mit besprochenen Verbesserungen aktualisiert und mit mehr Probanden wiederholt werden. Die Ergebnisse der Studie weisen darauf hin, dass Gamification die Motivation und vor allem "den Spaß" steigert, jedoch ist zu untersuchen ob Gamification immer helfen kann neue Softwaresysteme leichter zu erlernen.

# Bibliography

- [1] Tynan Sylvester. Designing games - a guide to engineering experiences. In *Designing Games - A Guide to Engineering Experiences*, pages 0–1000, Vancouver, Canada, Feb 2013.
- [2] Alexandre Thomas. Ganttproject. <https://www.ganttproject.biz/>, 2003.
- [3] J. Clement. Number of available applications in the google play store from december 2009 to june 2019. <https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/>, 2019.
- [4] Github. <https://github.com/>.
- [5] Christopher Hribar. Fork of ganttproject. <https://github.com/hribhrib/ganttproject>, 2019.
- [6] Ao.Univ.-Prof. Dr. Keith Andrews. 9.7.1 post-test interview. In *Human-Computer Interaction*, page 142, ISDS, Graz University of Technology, Inffeldgasse 16c, A-8010 Graz, Mar 2019.
- [7] Huotari, Kai, and Juho Hamari. "Defining gamification: a service marketing perspective." Proceeding of the 16th international academic MindTrek conference. ACM, 2012.
- [8] Esports Earnings. <https://www.esportsearnings.com/tournaments>
- [9] Hamari, Juho, Jonna Koivisto, and Harri Sarsa. "Does Gamification Work?-A Literature Review of Empirical Studies on Gamification." HICSS. Vol. 14. No. 2014. 2014.
- [10] StackOverflow, Badges. <https://stackoverflow.com/help/badges>
- [11] Gamification-Unternehmen. <https://www.gameeffective.com/>, <https://www.gamelearn.com/>, <https://www.gamewheel.com/>
- [12] Von gamewheel entwickelte Spiel für BurgerKing. <https://www.gamewheel.com/stories/burger-king>
- [13] GanttProject Tutorial Video, <https://www.youtube.com/watch?v=5rHCSa5ad34>

- [14] Muntean, Cristina. (2011). Raising engagement in e-learning through gamification. Proc. 6th International Conference on Virtual Learning ICVL.
- [15] Deterding, Sebastian, et al. "From game design elements to gamefulness: defining gamification." Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments. ACM, 2011.