

Hochschule für Technik und Wirtschaft Dresden

Fakultät Informatik/Mathematik

**Exemplarische Entwicklung von
Gamification-Elementen in der Lernplattform
ALADIN**

Bachelorarbeit

Zur Erlangung des akademischen Grades Bachelor of Science

eingereicht von: Alexander Schulz
Matrikelnummer: 49768
Studiengang: Wirtschaftsinformatik

eingereicht am:

Betreuer: Prof. Torsten Munkelt
Zweiter Gutachter: Paul Christ M. Sc.

Inhaltsverzeichnis

Inhaltsverzeichnis.....	I
Abbildungsverzeichnis.....	VI
Tabellenverzeichnis.....	IX
Glossar	X
1 Einleitung	1
1.1 Motivation - Problemstellung und Forschungsfragen	1
1.2 Zielsetzung	2
1.3 Forschungsmethoden und Aufbau der Arbeit	3
2 Grundlagen von Gamification.....	4
2.1 Definition und Abgrenzung von Gamification.....	4
2.2 Wirkungsweise von Gamification	4
2.2.1 Wirkung auf die Motivation.....	4
2.2.2 Wirkung auf die Leistung.....	6
2.2.3 Weitere theoretische Konzepte zur Wirksamkeit.....	7
2.3 Spielertypen und Gamification-Elemente	7
2.3.1 Spielertypen.....	7
2.3.2 Gamification-Elemente	9
2.3.3 Wirkung der Gamification-Elemente auf die Spielertypen.....	10
2.4 Aktuelle Anwendungsbereiche von Gamification	10
2.5 Positive und negative Effekte von Gamification.....	12
2.6 Erfolgsfaktoren von Gamification	13
3 Überblick über die Lernplattform ALADIN	14

3.1	Ziele der Lernplattform ALADIN	14
3.2	Hauptfunktionen und Merkmale der Lernplattform ALADIN.....	14
3.3	Benutzergruppen in der Lernplattform ALADIN.....	16
3.4	Die Software-Architektur der Lernplattform ALADIN	17
3.4.1	ALADIN-Frontend.....	17
3.4.2	ALADIN-Backend	17
4	Analyse der Anforderungen an Gamification in der Lernplattform ALADIN	19
4.1	Anforderungen der Lehrenden an Gamification in der Lernplattform ALADIN.	19
4.2	Anforderungen der Studierenden an Gamification in der Lernplattform ALADIN.....	19
4.3	Anforderungen der (Weiter-)Entwickler an Gamification in der Lernplattform ALADIN.....	19
5	Aktueller Stand der Technik in Bezug auf Gamification und ALADIN	20
5.1	Allgemeine Herangehensweisen bei der Entwicklung von Gamification.....	20
5.2	Technischer Zustand des ALADIN-Frameworks.....	22
5.3	Gamification-Frameworks und Gamification-Plattformen	27
5.4	Analyse der SkillTree Plattform zur Gamification der Lernplattform ALADIN.	28
6	Analyse der Integrationspunkte und relevanter Gamification-Elemente für die Lernplattform ALADIN	32
6.1	Analyse von Integrationspunkten für die inhaltliche Integration der Gamification in ALADIN.....	32
6.2	Analyse von Integrationspunkten für die technische Integration der Gamification in ALADIN.....	33

6.3	Identifizierung relevanter Gamification-Elemente für ALADIN.....	34
6.4	Bewertungsmatrix der Integrationspunkte und den relevanten Gamification-Elementen in ALADIN	36
7	Konzept(ion)/Entwurf der Gamification-Elemente für die Lernplattform ALADIN	39
7.1	Besonderheiten zur Konfiguration der Gamification-Elemente und Betrachtung von Folgefehlern.....	39
7.1.1	Konfiguration der Gamification-Elemente	39
7.1.2	Wertung von Folgefehlern	39
7.2	Betrachtung verschiedener Ansätze zur Integration des Punktesystems und der Serienbelohnung in ALADIN	40
7.2.1	Generelles Vorgehen bei allen Ansätzen	40
7.2.2	Erster Ansatz – Konvertierung der Schritte zu einem Objekt.....	40
7.2.3	Zweiter Ansatz – Erweiterung des Replays	42
7.2.4	Dritter Ansatz – Vergabe definierter Punktzahl pro Pfad	43
7.2.5	Vergleich der Ansätze	44
7.3	Integrationskonzept unter Verwendung der SkillTree Plattform	45
7.3.1	Entwurf der Schnittstelle zwischen ALADIN und der SkillTree Plattform	45
7.3.2	Entwurf der Skill-Struktur für die Verwendung von SkillTree mit ALADIN	46
7.3.3	Konzeption der Skill-Attribute.....	49
7.3.4	Entwurf der Level als Gamification-Element in ALADIN.....	50
8	Implementierung der Gamification-Elemente im ALADIN-Framework	52
8.1	Implementierungen der Services zur Verwendung der SkillTree Plattform	52

8.2	Implementierungen im Backend des ALADIN-Frameworks.....	55
8.3	Implementierung im Frontend des ALADIN-Frameworks	55
8.4	Herausforderungen bei der Implementierung der Gamification-Elemente	59
8.5	Endzustand der Infrastruktur zur Gamification der ALADIN Lernplattform unter Verwendung von SkillTree	60
9	Test der Gamification-Elemente im ALADIN-Framework.....	61
9.1	Test des Skill-Systems/Punktesystems als Gamification-Element im ALADIN- Framework.....	61
9.1.1	Testentwurf	61
9.1.2	Testdurchführung	61
9.1.3	Testauswertung	62
9.2	Test der Level als Gamification-Element im ALADIN-Framework	62
9.2.1	Testentwurf	62
9.2.2	Testdurchführung	62
9.2.3	Testauswertung	62
10	Fazit und Ausblick	63
10.1	Zusammenfassung der Ergebnisse	63
10.2	Beantwortung der Forschungsfragen.....	63
10.3	Implikation für die Praxis.....	64
10.4	Kritische Reflexion und Limitation der Arbeit	64
10.5	Ausblick und zukünftige Forschungen.....	64
11	Literaturverzeichnis.....	66
12	Eigenständigkeitserklärung	68
13	Anhang	69

13.1	Anforderungen an die Lernplattform ALADIN	69
13.2	Designprinzipien zum Einsatz von Gamification in der Hochschulbildung nach (Tolks & Sailer, 2021)	70
13.3	Einsatz von Gamification in unterschiedlichen Kontexten	72
13.4	Ablauf der Stücklistenauflösung in der ALADIN-Lernplattform	73
13.5	Anforderungen an die Gamification in der Lernplattform ALADIN – grafisch	76
13.6	Verknüpfung der Aufzeichnung mit den Skills durch die Gamificationdefinition	77
13.7	Test der Skill-Konfiguration	78
13.8	Test der Level-Konfiguration	79
13.9	Test der Gamificationdefinition	80
13.10	Test der Stücklistenauflösung	81
13.11	Test der Auftragsterminierung	83
13.12	Ergebnisse der Tests	84

Abbildungsverzeichnis

Abbildung 1: Selbstbestimmungstheorie nach Deci & Ryan (https://teufelsblog.files.wordpress.com/2021/04/image-1.png)	5
Abbildung 2: Bartle Spielertypen (Claus Brell, https://cbrell.de/blog/wp-content/uploads/2019/06/bartle-spielertypenbrell.png)	7
Abbildung 3: Technische Realisierung von ALADIN (Christ, et al., 2022).....	17
Abbildung 4: Schritte bei der Einführung von Gamification (Stieglitz, 2015).....	20
Abbildung 5: Aktueller Stand des ALADIN-Frameworks (Quelle: Eigene Darstellung)	22
Abbildung 6: Prozess der Aufgabendefinition durch Lehrende und Entwickler in ALADIN (Quelle: Eigene Darstellung)	23
Abbildung 7: Aufgabeninitialisierung und -bearbeitung durch Studierende in ALADIN (Quelle: Eigene Darstellung).....	25
Abbildung 8: Softwarearchitektur eines produktiven SkillTree Setups (https://skilltreeplatform.dev/assets/img/SkillsServiceArchitecture.89964c54.jpg).....	30
Abbildung 9: Gamificationdefinition als Schnittstelle zwischen SkillTree und ALADIN (Quelle: Eigene Darstellung).....	45
Abbildung 10: Java Struktur des SkillTree Imports (Quelle: Eigene Darstellung)	53
Abbildung 11: Programmierte Struktur der Gamificationdefinition im Frontend (Quelle: Eigene Darstellung).....	55
Abbildung 12: Endzustand der gamifizierten Lernplattform ALADIN (Quelle: Eigene Darstellung).....	60
Abbildung A13: Parametrisierung der Aufgabe (Quelle: Eigene Darstellung)	73
Abbildung A14: Wahl zwischen verschiedenen Lösungsvarianten (Quelle: Eigene Darstellung).....	73
Abbildung A15: Hinweise zum gewählten Lösungsweg (Quelle: Eigene Darstellung).74	

Abbildung A16: Ablesen der Adjazenzmatrix (Quelle: Eigene Darstellung).....	74
Abbildung A17: Vollständige und korrekte Adjazenzmatrix (Quelle: Eigene Darstellung)	75
Abbildung A18: Matrixmultiplikation anhand der Stücklistenauflösung in ALADIN (Quelle: Eigene Darstellung).....	75
Abbildung A19: Gamification requirements (Quelle: Eigene Darstellung)	76
Abbildung A20: Verbindung der Aufzeichnung mit den Skills - Gamificationdefinition (Quelle: Eigene Darstellung).....	77
Abbildung A21: Definition des Moduls Produktionswirtschaftslehre (Quelle: Eigene Darstellung).....	78
Abbildung A22:Definition des Moduls Informationsmanagement (Quelle: Eigene Darstellung).....	78
Abbildung A23: Definition der Level auf ALADIN-Ebene (Quelle: Eigene Darstellung)	79
Abbildung A24: Definition der Level auf Studienmodulebene (Quelle: Eigene Darstellung).....	79
Abbildung A25: Definition der Pfad-Skill Verknüpfung der Stücklistenauflösungsaufgabe (Quelle: Eigene Darstellung).....	80
Abbildung A26: Definition der Pfad-Skill Verknüpfung der Auftragsterminierungsaufgabe (Quelle: Eigene Darstellung).....	80
Abbildung A27: Erfolgreiche Punktevergabe bei der Parametrisierung der Stücklistenauflösungsaufgabe (Quelle: Eigene Darstellung).....	81
Abbildung A28: Keine Punktevergabe bei der Parametrisierung der Stücklistenauflösungsaufgabe, da Maximum der Lerneinheit erreicht (Quelle: Eigene Darstellung).....	81
Abbildung A29: Erfolgreiche Punktevergabe beim Ablesen der Adjazenzmatrix mit Hilfe (Quelle: Eigene Darstellung).....	81

Abbildung A30: Erfolgreiche Punktevergabe bei der Direktbedarfsmatrixmultiplikation mit Hilfe (Quelle: Eigene Darstellung)	82
Abbildung A31: Erfolgreiche Punktevergabe beim Ablesen der Adjazenzmatrix (Quelle: Eigene Darstellung).....	82
Abbildung A32: Erfolgreiche Punktevergabe bei der Direktbedarfsmatrixmultiplikation (Quelle: Eigene Darstellung).....	82
Abbildung A33: Erfolgreiche Punktevergabe bei der Parametrisierung der Auftragsterminierungsaufgabe (Quelle: Eigene Darstellung).....	83
Abbildung A34: Erfolgreiche Punktevergabe bei der Durchführung der Metra-Potenzial-Methode (Quelle: Eigene Darstellung)	83
Abbildung A35: Erfolgreiche Punktevergabe beim Durchführen der Zeitanalyse (Quelle: Eigene Darstellung).....	83
Abbildung A36: Initiale Skill-Ansicht zu Beginn des Tests (Quelle: Eigene Darstellung)	84
Abbildung A37: Finale Übersicht aller Punkte und Studienmodule nach dem Test (Quelle: Eigene Darstellung).....	84
Abbildung A38: Finale Übersicht aller Punkte der Produktionswirtschaftslehre nach dem Test (Quelle: Eigene Darstellung).....	84

Tabellenverzeichnis

Tabelle 1: Spiel-Design-Elemente, deren ausgelöste Mechanismen und die hierdurch erfüllten psychologischen Grundbedürfnisse (Sailer, 2016)	6
Tabelle 2: Gamification-Elemente und deren Wirkung (Quelle: Eigene Tabelle)	9
Tabelle 3: Zuordnung Gamification-Elemente zu den Spielertypen nach Bartle	10
<i>Tabelle 4: Vergleich ausgewählter Gamification Lösungen (Quelle: Eigene Tabelle) ..</i>	<i>27</i>
Tabelle 5: Vergleich zwischen der Integration im Frontend und Backend (Quelle: Eigene Tabelle)	33
Tabelle 6: Bewertung der inhaltlichen Integrationspunkte und den relevanten Gamification-Elementen in ALADIN (Quelle: Eigene Darstellung)	36
Tabelle 7: Bewertung der technischen Integrationspunkte den relevanten Gamification-Elementen in ALADIN (Quelle: Eigene Darstellung)	37
Tabelle 8: Vergleich der Ansätze zur Integration der Gamification-Elemente in ALADIN (Quelle: Eigene Darstellung)	44
Tabelle 9: Mögliche Skill-Strukturen für die Abbildung der Aufgaben aus ALADIN (Quelle: Eigene Darstellung)	46
Tabelle 10: Auswirkung der Hilfsmethoden auf die Punktzahl (Quelle: Eigene Tabelle)	49
Tabelle A11: Designprinzipien zum Einsatz von Gamification in der Hochschulbildung (Quelle: Eigen Tabelle)	70
Tabelle A12: Übersicht Kontexte empirischer Studien (Sailer, 2016)	72

Glossar

Auftragsterminierung:

Auftragsterminierung ist ein Konzept zur effizienten Planung und Koordination von Produktionsprozessen. Es umfasst die Verwaltung von Aufträgen, Ressourcen, Abläufen und Terminen, um die Produktion zu optimieren.

Backend:

Das Backend bezieht sich auf den nicht sichtbaren Teil einer Software, der für die Verarbeitung von Daten, Geschäftslogik und Kommunikation mit den Clients und Services verantwortlich ist.

Bootstrap:

Bootstrap ist ein beliebtes CSS-Framework, das vorgefertigte Designs und Stile für die Gestaltung von Websites und Webanwendungen bietet.

Client Credential Flow:

Der Client Credential Flow ist ein OAuth2-Fluss, der es einem Client (normalerweise einem Server) ermöglicht, ohne Benutzerinteraktion einen Access Token zu erhalten.

Event-Broker:

Ein Event-Broker ist eine Softwarekomponente oder ein Dienst, der in einem System eingesetzt wird, um die Verteilung und Weiterleitung von Ereignissen oder Nachrichten zwischen verschiedenen Teilen des Systems zu ermöglichen.

Framework

Ein Framework ist eine vorstrukturierte Sammlung von Tools, Bibliotheken und Standards, die Entwicklern als Grundlage für die effiziente Gestaltung von Softwareanwendungen dient. Es erleichtert die Entwicklung, fördert die Wiederverwendung von Code und folgt bewährten Praktiken.

Frontend:

Das Frontend ist der sichtbare Teil einer Software, den die Benutzer interaktiv verwenden. Es umfasst die Benutzeroberfläche, mit der die Benutzer interagieren und die Präsentation von Daten.

Java-Spring-Boot:

Spring Boot ist ein Framework für die Entwicklung von Java-Anwendungen. Es erleichtert die Konfiguration und Erstellung von produktionsfertigen Anwendungen.

JSON:

JSON (JavaScript Object Notation) ist ein leicht lesbares Datenformat, das häufig zur Übertragung von strukturierten Daten zwischen einem Server und einem Webbrowser verwendet wird.

Modellierungsaufgaben

Modellierungsaufgaben, häufig aus dem Bereich der Mathematik bekannt, beschreiben einen realen Sachverhalt und zielen darauf ab, die wesentlichen Eigenschaften und Zusammenhänge dieses Sachverhalts abzubilden. Diese Aufgaben führen zu einem besseren Verständnis und zur Berufsvorbereitung bei den Lernenden, da mehrere Schritte zum Lösen durchgeführt werden müssen (Blum, 2007).

Node.js

Node.js ist eine serverseitige JavaScript-Laufzeitumgebung, die auf der V8 JavaScript-Engine von Google basiert. Sie ermöglicht die Ausführung von JavaScript-Code außerhalb eines Webbrowsers und wird häufig für die Entwicklung von skalierbaren Netzwerkanwendungen eingesetzt.

(JSON/Javascript-)Objekt:

Ein Objekt im Kontext dieser Arbeit ist eine Sammlung von Schlüssel-Wert-Paaren, bei dem über den zugehörigen Schlüssel auf den Wert zugegriffen werden kann.

OAuth:

OAuth ist ein Autorisierungsprotokoll, das es Benutzern ermöglicht, Dritten eingeschränkten Zugriff auf ihre Ressourcen zu gewähren, ohne ihre Anmeldeinformationen weiterzugeben.

Queues:

Im Kontext von Systemarchitektur und Kommunikationstechnologie bezieht sich "Queue" auf eine Datenstruktur, die Elemente in einer bestimmten Reihenfolge speichert und abrufen. Es arbeitet nach dem Prinzip "First In, First Out" (FIFO), was bedeutet, dass

das zuerst eingefügte Element auch als erstes entfernt wird. Queues werden häufig verwendet, um Aufgaben zu verwalten, Nachrichten zu puffern oder die Kommunikation zwischen verschiedenen Teilen eines Systems zu steuern.

RabbitMQ Stomp-Broker:

Der RabbitMQ Stomp-Broker ist eine Integration von STOMP (Simple Text Oriented Messaging Protocol) in RabbitMQ. STOMP ist ein textbasiertes Kommunikationsprotokoll, das die Nachrichtenübermittlung zwischen Anwendungen erleichtert. Diese Integration ermöglicht es Anwendungen, effizient mit RabbitMQ über STOMP zu kommunizieren, was eine standardisierte und effektive Art des Nachrichtenaustauschs in verteilten Systemen ermöglicht

Regex/Reguläre Ausdrücke:

Reguläre Ausdrücke (Regex) sind Muster, die verwendet werden, um Zeichenfolgen in Texten zu suchen, zu manipulieren oder zu bearbeiten. Sie bieten eine flexible Möglichkeit, komplexe Such- und Ersetzungsvorgänge durchzuführen.

Remote Procedure Calls (RPC):

Remote Procedure Calls (RPC) sind eine Technik in der Informatik, die es ermöglicht, dass eine Computerprogrammkomponente (z.B. in einer Client-Server-Architektur) eine Prozedur oder Funktion in einem anderen Adressraum (häufig auf einem entfernten Rechner) aufruft. Dies erleichtert die Interprozesskommunikation und ermöglicht die Ausführung von Funktionen auf entfernten Rechnern, als ob sie lokal aufgerufen würden.

Session-Cookie:

Ein Session-Cookie ist eine Art von Cookie, das nur für die Dauer der Benutzersitzung aufbewahrt wird. Es wird normalerweise gelöscht, wenn der Browser geschlossen wird.

Softwarearchitektur:

Die Softwarearchitektur beschreibt die grundlegende Struktur und Organisation einer Softwareanwendung. Sie definiert die Komponenten, ihre Interaktionen, Muster und Prinzipien, die für das Design und die Entwicklung der Software gelten.

Spaced-Repetition:

Spaced-Repetition ist eine Lernmethode, bei der Inhalte in bestimmten Zeitabständen

wiederholt werden, um das langfristige Gedächtnis zu verbessern. Die Abstände zwischen den Wiederholungen nehmen mit jeder erfolgreichen Erinnerung zu.

Stücklistenauflösung:

Die Stücklistenauflösung bezieht sich auf die Analyse und Aufschlüsselung von komplexen Strukturen, typischerweise in der Fertigungsindustrie. Sie zeigt die Hierarchie und Beziehungen zwischen verschiedenen Elementen auf, ähnlich einem Gozintographen, um eine bessere Übersicht über die Zusammensetzung zu erhalten.

TypeScript:

TypeScript ist eine Programmiersprache, die auf JavaScript aufbaut und zusätzliche Typisierung bietet, um die Entwicklung robusterer Anwendungen zu unterstützen.

Vue.js

Vue.js ist ein populäres open-source JavaScript-Framework zur Entwicklung von flexiblen Benutzeroberflächen und Single-Page-Anwendungen. Vue.js bietet eine reaktive Datenbindung, Komponentenarchitektur und vieles mehr, was es zu einer beliebten Wahl für Webentwickler macht.

WebSockets

WebSocket ist ein fortgeschrittenes Netzwerkprotokoll, welches eine bidirektionale Kommunikation zwischen einem Webbrowser und einem Webserver in Echtzeit ermöglicht. Dadurch ermöglichen es die nahtlose und kontinuierliche Übertragung von Daten zwischen Client und Server, was besonders in Anwendungen relevant ist, die Echtzeit-Updates oder schnelle Datenübertragungen erfordern. Werden mehrere WebSocket Verbindungen aufgebaut, bezeichnet man diese als WebSockets.

Worker:

Ein Worker ist ein Softwaremodul, das innerhalb dieser Architektur spezielle Aufgaben abarbeitet. Im ALADIN Kontext hat jeder Aufgabentyp seinen eigenen Worker. Diese Worker übernehmen Aufgaben, die asynchron und in Reaktion auf eingehende Events von den Nutzerinteraktionen ausgelöst werden. Zum Beispiel könnte ein Worker dafür zuständig sein, eine konkrete Aufgabenstellung anhand der eingegebenen Parameter zu generieren.

1 Einleitung

1.1 Motivation - Problemstellung und Forschungsfragen

Die rasante Entwicklung digitaler Technologien und die plötzliche Corona-Pandemie hatten einen signifikanten Einfluss auf die weltweite Bildungslandschaft (Wilmers, et al., 2021). Zentrale Lernplattformen sind aus dem heutigen Bildungsprozesse nicht mehr wegzudenken und bieten den Lernenden eine Vielzahl von Möglichkeiten, ihre Kenntnisse und Fähigkeiten zu erweitern. Zusätzlich bieten sie den Lehrenden eine Möglichkeit, die Studierenden besser in den Prozess einzubinden. Eine der Herausforderungen für Bildungseinrichtungen besteht darin, die Lernmotivation der Studierenden aufrechtzuerhalten und ihre aktive Teilnahme am Lernprozess zu steigern. In diesem Zusammenhang hat die Gamification als vielversprechender Ansatz große Aufmerksamkeit erlangt. Durch die Integration von spielerischen Elementen in Bildungsplattformen, können Studierende motiviert werden, ihr Engagement und ihre Leistung zu steigern (Pawelka, et al., 2017). Diese Ansicht wird durch die Tatsache unterstützt, dass 88 % der 16-29-Jährigen zumindest hin und wieder Computer- oder Videospiele spielen, was die Attraktivität von gamifizierten Lernansätzen in dieser Altersgruppe unterstreicht (Bitkom, 2022).

Die vielversprechenden Potenziale von Gamification in der Bildung sind der Grund dafür, die konkrete Entwicklung und Implementierung von Gamification-Elementen in der Lernplattform ALADIN darzustellen. Obwohl Gamification als ein effektiver Ansatz zur Steigerung der Lernmotivation und des Lernverhaltens gilt, gibt es immer noch nur eine begrenzte Anzahl von Studien, die sich mit der praktischen Umsetzung in Bildungskontexten befassen. Insbesondere fehlt es an umfassenden Untersuchungen, die den gesamten Prozess der Entwicklung, die spezifischen Elemente und ihre Auswirkungen auf die Studierenden umfassend analysieren.

Die zentrale Forschungsfrage dieser Arbeit lautet daher:

„In welcher Weise können Gamification-Elemente in der Lernplattform ALADIN entwickelt und implementiert werden, um die Lernmotivation und das Lernverhalten der Studierenden nachhaltig zu verbessern?“

Daraus ergeben sich folgende Unterfragen:

1. *„Welche Erfolgsfaktoren gibt es bei der Konzeption von Gamification-Elementen?“*
2. *„Welche Ansätze eignen sich für die Implementierung der Gamification-Elemente in ALADIN?“*
3. *„Welche Herausforderungen gibt es bei der Implementierung der Gamification-Elemente?“*

Die Beantwortung dieser Fragen wird es ermöglichen, die erfolgreichsten Ansätze und Strategien zu identifizieren, die sich für die Entwicklung und Implementierung von Gamification-Elementen in Lernplattformen eignen. Durch das Verständnis der effektivsten Vorgehensweisen können Bildungseinrichtungen gezielt auf die Bedürfnisse der Lernenden eingehen und ihre Lernplattformen optimieren, um eine nachhaltige Verbesserung der Lernmotivation und des Lernverhaltens zu erreichen.

1.2 Zielsetzung

Die Funktionen von ALADIN gehen über die einer herkömmlichen Lernplattform hinaus. Als innovatives Framework ist ALADIN nicht nur eine statische Plattform zum Lernen, sondern generiert zufallsbasierte Aufgaben und interaktive Lösungshilfen. Um die Lernumgebung noch nachhaltiger und effektiver zu gestalten, sollen verschiedene Gamification-Elemente entwickelt werden. Diese sollen die Motivation der Lernenden erhöhen und somit zu einer Optimierung des Lernprozesses führen. Zu beachten ist, dass die Gamification-Elemente nicht fest auf bestimmte Aufgaben, wie dem Definieren von SQL-Statements, zugeschnitten werden sollen, sondern deklarativ an dahinter liegende Basiselemente geknüpft werden.

Die Implementierung dieser flexiblen Gamification-Elemente erfordert einen sehr generischen Ansatz und fundiertes Verständnis der dahinterliegenden Techniken. Nur dadurch kann eine geringe Kopplung der Elemente an die verschiedenen Aufgabentypen gewährleistet werden. Dazu gehören beispielsweise Algorithmen zur Bewertung von Leistungen, Belohnungssysteme und Fortschrittsverfolgung.

1.3 Forschungsmethoden und Aufbau der Arbeit

In der Arbeit wird zuerst auf den theoretischen Rahmen von Gamification eingegangen. Hierbei werden bereits vorhandene Erkenntnisse analysiert und der aktuelle Stand der Forschung erläutert. Dadurch wird ein Überblick zu verschiedenen Konzepten und Theorien von Gamification hergestellt.

Außerdem wird mit einer Fallstudie die konkrete Implementierung von Gamification-Elementen in ALADIN untersucht. Hierbei werden relevante Gamification-Elemente identifiziert und an geeigneten Stellen in die Lernplattform integriert. Die Auswahl der Elemente sollte auf den Zielen der Lernplattform und den Bedürfnissen der Nutzenden basieren. Im Laufe des Entwurfs und der Konzeption werden Besonderheiten bei der Implementierung betrachtet. Die Durchführung dieser Fallstudie ermöglicht es, konkrete Einblicke in die Implementierung von Gamification-Elementen in ALADIN zu gewinnen. Diese Erkenntnisse können verwendet werden, um ähnlich gestaltete Systeme zu gamifizieren.

2 Grundlagen von Gamification

2.1 Definition und Abgrenzung von Gamification

Definitionen für Gamification gibt es viele, was zeigt, dass es weiterhin viel Dynamik zu diesem Konzept gibt. Dennoch haben die meisten Definitionen eines gemein: spielerische Elemente werden gezielt in nicht-spielbezogenen Kontexten eingesetzt. Dadurch sollen Anwender einen zusätzlichen Anreiz bekommen, bestimmte Handlungen auszuführen (Sailer, 2016). Der Einsatz von Erfahrungspunkten, Herausforderungen (Quests) oder Punktesystemen kann nachweislich die Motivation der Nutzer steigern und zu mehr Interaktionen führen (Sailer, 2016).

Gamification sollte nicht mit „Serious Gaming“ verwechselt werden, dieses verwandte Konzept beschäftigt sich mit Spielen, welche nicht ausschließlich zur Unterhaltung dienen, sondern zusätzlich einen lehrenden Charakter haben. So können bei Flugsimulatoren Landemanöver aus der Realität abbilden, wodurch Lerneffekte erzielt werden (Stieglitz, 2015).

2.2 Wirkungsweise von Gamification

2.2.1 Wirkung auf die Motivation

Insbesondere die Motivation und das Engagement sind zentrale Faktoren in Lernprozessen (Sailer, 2016). Die Einflüsse von Gamification auf diese Aspekte lassen sich gut durch die Selbstbestimmungstheorie erklären. Nach dieser gibt es drei Grundbedürfnisse: das Bedürfnis nach Selbstbestimmung, nach Verbundenheit mit anderen und das Bedürfnis nach Kompetenz (Ryan & Deci, 2000).



Abbildung 1: Selbstbestimmungstheorie nach Deci & Ryan (<https://teufelsblog.files.wordpress.com/2021/04/image-1.png>)

Vor allem aber intrinsische Bedürfnisse, wie das Streben nach Wachstum, sozialem Austausch und dem Wissen etwas Nützliches zu tun, können durch Gamification-Elemente befriedigt werden (Stieglitz, 2015; Alsawaier, 2017; Sailer, 2016).

Tabelle 1: Spiel-Design-Elemente, deren ausgelöste Mechanismen und die hierdurch erfüllten psychologischen Grundbedürfnisse (Sailer, 2016)

Spiel-Design-Element	Mechanismus	Psychologisches Grundbedürfnis
Punkte	Granulares Kompetenzfeedback	Kompetenzerleben
Leistungsgraphen	Nachhaltiges Kompetenzfeedback	Kompetenzerleben
Abzeichen	Kumulatives Kompetenzfeedback	Kompetenzerleben
(Team-)Bestenlisten	Kumulatives Kompetenzfeedback	Kompetenzerleben
Avatar	Wahlmöglichkeiten	Autonomieerleben (Entscheidungsfreiheit)
Narrativ	Volitionales Engagement	Autonomieerleben (Aufgabenbedeutsamkeit)
Narrativ	Momente der Relevanz	Erleben sozialer Eingebundenheit
Narrativ	gemeinsames Ziel	Erleben sozialer Eingebundenheit
Team-Bestenlisten	gemeinsames Ziel	Erleben sozialer Eingebundenheit
(Team-)Bestenlisten	Konstruktiver Wettbewerb	Erleben sozialer Eingebundenheit

Die Tabelle 1 stellt die Wirkungsweise von verschiedenen Gamification-Elementen (hier Spiel-Design-Elemente) dar. Zusätzlich werden ausgelöste Mechanismen und die damit in Zusammenhang hängenden Grundbedürfnisse abgebildet.

Es gibt in der Wissenschaft viele verschiedene Ansätze, um die Wirkung von Gamification auf die Motivation zu begründen. Gerade die Wirkung der Gamification-Elemente auf die psychologischen Grundbedürfnisse verdeutlicht den Nutzen von Gamification.

2.2.2 Wirkung auf die Leistung

Grundsätzlich wird die Leistung durch verschiedene Faktoren beeinflusst, vor allem Personenfaktoren und Umweltfaktoren spielen eine entscheidende Rolle (Sailer, 2016). Viele Studien zeigen, dass Gamification zu einer Leistungsförderung führen kann (Sailer, 2016). Ursachen dafür könnten Umweltfaktoren, wie spezifische Zielsetzungen durch ein

Punktesystem oder Herausforderungen sein. Entscheidend ist allerdings ein geeignetes Feedbacksystem, welches durch Bestenlisten abgebildet werden kann. Die Personenfaktoren stehen in enger Verbindung zu der in 2.2.1 betrachteten Selbstbestimmungstheorie, mithilfe welcher die Motivationssteigerung begründet werden kann (Sailer, 2016).

2.2.3 Weitere theoretische Konzepte zur Wirksamkeit

Doch nicht nur die Selbstbestimmungstheorie könnte eine Begründung für die Wirksamkeit von Gamification sein. Andere theoretische Konzepte sind beispielsweise die Bedürfnispyramide von Maslow, die ERG-Theorie von Alderfer, die Zielsetzungstheorie von Locke und Latham und die Flow-Theorie von Csikszentmihalyi (Stieglitz, 2015). Eine genaue Trennung zwischen Elementen, welche nur die Motivation steigern und Elementen, die nur die Leistung steigern, kann nicht durchgeführt werden, da die Elemente meistens mehrere Faktoren simultan beeinflussen (Sailer, 2016).

2.3 Spielertypen und Gamification-Elemente

2.3.1 Spielertypen

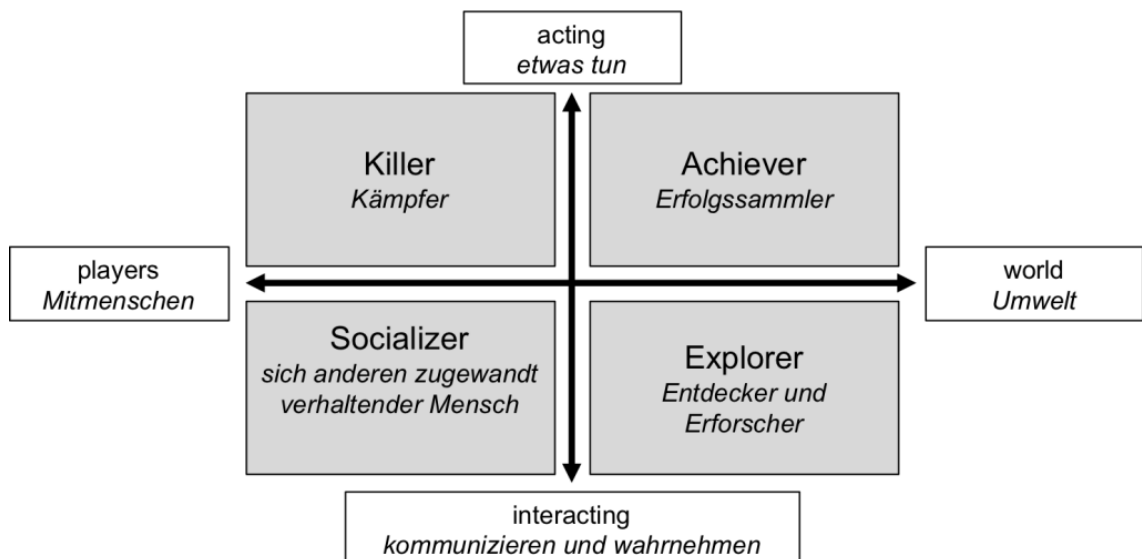


Abbildung 2: Bartle Spielertypen (Claus Brell, <https://cbrell.de/blog/wp-content/uploads/2019/06/bartle-spielertypenbrell.png>)

Verschiedene Nutzer reagieren anders auf Gamification und haben andere Ziele. Dies sagt auch die Spielertypen-Taxonomie von Bartle. Er unterscheidet zwischen vier

Spielertypen (Bartle, 1996):

1. Killer: Möchten andere Spieler besiegen und ihre Überlegenheit demonstrieren
2. Achiever: Möchten im Spiel Erfolge sammeln und hohe Punktzahlen erreichen
3. Socialiser: Legen besonderen Wert auf soziale Interaktion und Kommunikation
4. Explorer: Möchten die Spielwelt erkunden und neue Dinge entdecken

Bei der Auswahl und Implementierung der Gamification-Elemente sollte explizit darauf geachtet werden, alle Spielertypen und deren individuelle Anforderungen zu beachten.

2.3.2 Gamification-Elemente

Tabelle 2: Gamification-Elemente und deren Wirkung (Quelle: Eigene Tabelle)

Element	Beschreibung	Wirkung
Punktesystem, Level (Sailer, 2016)	Sammeln von Punkten für erledigte Aufgaben, Level basierend auf Punktestand	Motivation, Gefühl des Fortschritts
Serienbelohnung (Munkelt & Christ, 2022)	Zusatzpunkte, wenn Aufgaben hintereinander (ohne Unterbrechung) gelöst werden	Motivation, Vertiefung
Badges/Auszeichnungen (Sailer, 2016)	Auszeichnungen für bestimmte Leistungen	Anerkennung, Motivation
Fortschrittsbalken (Korn, et al., 2022)	Visualisierung des Lernfortschritts	Transparenz, motivierende Ziele
Herausforderungen/Quests (Sailer, 2016)	Definition von gebündelten Aufgaben mit klarem Ziel	Motivation, Wahlmöglichkeit, gezieltes Aneignen von Fähigkeiten
Foren, Hilfesuche, Chats (Korn, et al., 2022)	Interaktion zwischen Nutzern, Hilfe wird angefordert	Gemeinschaft, Engagement
Avatar (Sailer, 2016)	Visuelle Repräsentation des Nutzers	Identifikation, Selbstbestimmung
Ranglisten (Sailer, 2016)	Vergleich der Leistung mit anderen	Wettbewerb, Motivation

Tabelle 2 stellt eine Auswahl häufig verwendeter Gamification-Elemente und deren Wirkung dar. Die Auswahl stammt aus einer Analyse mehrerer wissenschaftlicher Quellen, welche im Rahmen dieser Arbeit durchgeführt wurde. Im Laufe der Arbeit wird genauer auf die jeweiligen Elemente eingegangen und gezeigt, wie eine engere Auswahl in ALADIN integriert werden kann.

2.3.3 Wirkung der Gamification-Elemente auf die Spielertypen

Punktesysteme und Level sprechen insbesondere Achiever an, da diese kontinuierliche Erfolge durch das Sammeln von Punkten und Aufstieg in den Leveln erleben möchten.

Badges in Form von Auszeichnungen für bestimmte Leistungen motivieren ebenfalls Achiever. Besonders herausfordernde oder versteckte Badges können aber auch Explorer ansprechen, die gerne Neues entdecken.

Für Explorer könnte es zusätzlich spezielle Badges geben, die nur durch das horizontale Lösen vieler unterschiedlicher Aufgaben erreicht werden können.

Social Gamification in Form von Foren und anderen Interaktionsmöglichkeiten sprechen vor allem Socialiser an, welche Wert auf Kommunikation legen.

Ranglisten und andere Möglichkeiten zum Leistungsvergleich sprechen die wettbewerbsorientierten Killer an.

Tabelle 3: Zuordnung Gamification-Elemente zu den Spielertypen nach Bartle

Gamification-Element	Angesprochene Spielertypen
Punktesystem, Level	Achiever
Badges	Achiever, Explorer
Forum, Hilfesuche	Socialiser
Ranglisten	Killer

Insgesamt sollten die Elemente so kombiniert werden, dass alle Typen angesprochen werden. Dabei sind auch individuelle Anpassungsmöglichkeiten wichtig, damit sich jeder Nutzer auf seine bevorzugten Elemente konzentrieren kann (Sailer, 2016).

2.4 Aktuelle Anwendungsbereiche von Gamification

In der heutigen Zeit wird Gamification nicht nur in der Bildung, sondern auch in einer Vielzahl anderer Bereiche verstärkt eingesetzt (Sailer, 2016). Um die Vielfalt der Anwendungsfelder und deren unterschiedliche Zielsetzungen zu beleuchten, wird in dieser Arbeit eine umfassende Analyse von 56 Studien herangezogen. Diese Analyse wurde von Sailer durchgeführt. Die Tabelle A12 im Anhang zeigt eine Übersicht zu den verschiedenen Anwendungsbereichen von Gamification. Die Arbeit von Sailer dient als

wertvolle Ressource, um die breite Palette der Anwendungen von Gamification besser zu verstehen.

In Bezug auf die Anzahl der durchgeführten Studien stellt der Bildungsbereich den am meisten erforschten Anwendungskontext dar. 17 Studien können diesem Bereich zugeordnet werden. Hierbei handelt es sich um Gamification-Anwendungen im schulischen oder hochschulischen Umfeld, um den Lernerfolg und die Motivation zu steigern. Der Arbeitskontext umfasst neun der untersuchten Studien, die entweder die Gamification innerbetrieblicher Interaktionen und Kommunikation oder die Integration von Gamification in Arbeitsprozessen untersuchen. Ersteres soll die Vernetzung, Kollaboration und Partizipation fördern, wohin gegen zweiteres die Qualität, Motivation und Leistung optimieren soll.

Weitere Bereiche und deren Ziele sind:

1. Online-Communities und Soziale Netzwerke:

Ziel: Steigerung der Nutzeraktivität und Engagement in Online-Communities und sozialen Netzwerken durch gamifizierte Elemente.

2. Gesundheitssektor:

Ziel: Unterstützung therapeutischer Prozesse, Förderung von sportlicher Betätigung und Ernährungsumstellungen durch Integration von Gamification zur Steigerung der Motivation und Adhärenz.

3. Crowdsourcing:

Wirtschaftlicher Fokus: Verbesserung der Qualität und Leistung bei Crowdsourcing-Tätigkeiten durch gamifizierte Anreize.

Wissenschaftlicher Fokus: Steigerung der Motivation für Forschungstätigkeiten im Bereich Crowdsourcing.

Gamification findet in verschiedenen Kontexten Anwendung und zielt im Allgemeinen darauf ab, Ziele auf Erlebensebene (Motivation, Freude) und Verhaltensebene (Vernetzung, Partizipation, Leistung, Genauigkeit) zu erreichen. In verschiedenen Anwendungskontexten wie Arbeitsumgebungen, Bildung, Crowdsourcing, Datenerhebungen, Gesundheitswesen, Marketing, Online-Communities und Umweltschutz stehen jeweils unterschiedliche Aspekte im Fokus, jedoch bleibt die

Betonung auf Motivation und Verhaltensänderung auf beiden Ebenen (Sailer, 2016).

2.5 Positive und negative Effekte von Gamification

Beim Einsatz von Gamification sind neben den häufig positiven Resultaten auch zahlreiche Fälle aufgetreten, in denen entweder gemischte Ergebnisse festgestellt wurden, die sowohl positive als auch negative Effekte umfassen als auch rein negative Resultate (Stieglitz, 2015). Deshalb wird der Einsatz von Gamification mitunter kritisch bewertet (Borukhovich-Weis, et al., 2021). Im Folgenden werden einige Effekte dargestellt:

Als negative Effekte werden unter anderem genannt:

- Druck oder das Gefühl bewertet zu werden (Borukhovich-Weis, et al., 2021)
- Unsicherheit und Neid (Tolks & Sailer, 2021)
- kein konkreter Nutzen für die Nutzer*innen (Tolks & Sailer, 2021)
- Wettkämpfe können zu hohem Stresslevel führen (Tolks & Sailer, 2021)

Folgende positive Effekte wurden genannt:

- lockerer Ansatz beim Lösen von Aufgaben (Borukhovich-Weis, et al., 2021)
- Steigerung des Engagements der Studierenden durch schnelles Feedback (Tolks & Sailer, 2021)
- konstruktiver Wettbewerb fördert die Motivation (Tolks & Sailer, 2021)
- Wettkampf ist fördernder, wenn alle Lernenden auf dem gleichen Leistungsniveau sind (Tolks & Sailer, 2021)

Diese Ergebnisse bestärken die vorherigen Aussagen, dass eine überlegte Auswahl und Konzeption der Gamification-Elemente ausschlaggebend für den Erfolg ist.

2.6 Erfolgsfaktoren von Gamification

Basierend auf den vorherigen Kapiteln ergeben sich folgende Erfolgsfaktoren:

Passende Kombination von Gamification-Elementen für die Zielgruppe:

Die Elemente sollten so gewählt werden, dass sie die Motivation der jeweiligen Nutzer ansprechen. Dazu ist es hilfreich, die Spielertypen nach Bartle zu berücksichtigen und eine Mischung aus Elementen für Achiever, Explorer, Socialiser und Killer einzusetzen.

Konkreter Nutzen für die Zielgruppe:

Gamification sollte nicht um ihrer selbst Willen eingesetzt werden, sondern einen konkreten Zweck für die Nutzer*innen erfüllen, wie z.B. Lernfortschritt oder gesundheitsförderndes Verhalten.

Vermeidung von Überforderung:

Zu viel Wettbewerbsdruck durch Ranglisten kann Stress auslösen. Daher sollte auf ein angemessenes Maß geachtet werden. Auch insgesamt sollte Gamification motivierend wirken und keine Überforderung erzeugen.

Passendes Feedbacksystem:

Regelmäßiges Feedback zum Fortschritt, z.B. durch Punkte oder Level ist wichtig, um die positiven Effekte zu erzielen.

Freiwillige Teilnahme:

Druck sollte vermieden werden. Die Nutzer sollten selbst entscheiden können, in welchem Umfang sie die Gamification-Elemente nutzen möchten.

Datenschutz:

Bei Gamification fallen mögliche persönliche Daten der Nutzer an. Der Schutz dieser Daten sollte sichergestellt sein.

Evaluierung der Wirkung:

Um die konkreten Auswirkungen zu überprüfen, sollten Evaluationen durchgeführt werden. So können gegebenenfalls Anpassungen vorgenommen werden.

3 Überblick über die Lernplattform ALADIN

ALADIN: Generator für Aufgaben und Lösung(shilfen) Aus Der Informatik und angrenzenden Disziplinen

Der gegenwärtige Hauptentwickler von ALADIN ist Paul Christ M. Sc., er hat maßgeblich zur Entwicklung des Systems beigetragen, einschließlich vieler technischer Konzepte.

3.1 Ziele der Lernplattform ALADIN

- Zeit- und ortsunabhängiges Lernen soll ermöglicht werden
- Selbständiges und individuelles Üben der Studierenden ermöglichen
- Reduzierung der Bindung von Lehrpersonal bei der Bereitstellung von Aufgaben und Lösungshilfen
- Entlastung des Lehrpersonals bei der Korrektur von Lösungen
- Ermöglichung eines asynchronen Austauschs zwischen Studierenden und Lehrpersonal über geteilte Lösungsversuche
- Anpassung der Komplexität der Aufgaben an den Kenntnisstand der Lernenden
- Automatische und sofortige Rückmeldung über die Korrektheit von Lösungen
- Nutzung der Lernplattform zur Erstellung individueller Prüfungsaufgaben durch das Lehrpersonal
- Vertiefung des Verständnisses durch schrittweise Lösungshilfen und Wiedergabe von Lösungsversuchen

(Christ, et al., 2022)

3.2 Hauptfunktionen und Merkmale der Lernplattform ALADIN

Das manuelle Erstellen von Modellierungsaufgaben ist in der Regel sehr zeitaufwändig und anspruchsvoll. Daher stehen oft nur begrenzt Übungsaufgaben zur Verfügung.

Zudem ist es meist so, dass Lernende häufig Fragen zum Lösungsweg haben oder Unterstützung bei der Lösungsfindung benötigen. Das bereitet dem Lehrpersonal erheblichen Aufwand und ist außerhalb der Lehrveranstaltungen schwer umsetzbar, da die Verfügbarkeit des Lehrpersonals knapp ist (Christ, et al., 2022).

Daraus ergeben sich für ALADIN folgende Hauptfunktionen:

1. Automatische Generierung von Modellierungsaufgaben sowie entsprechenden Lösungshilfen und Hinweisen
2. Parametrisierbare und zufallsbasierte Erzeugung von Aufgaben, um die Komplexität an den Lernstand der Studierenden anzupassen
3. Zeit- und ortsunabhängige Bereitstellung der Aufgaben über das Internet
4. Unterstützung beim Lösen durch schrittweise Teillösungen und Tipps
5. Aufzeichnung und Analyse von Lösungsversuchen
6. Automatische Korrektur der Lösungen
7. Reduzierung der Bindung von Lehrpersonal

(Christ, et al., 2022)

Bisher wurden Modellierungsaufgaben für folgende Aufgaben umgesetzt:

1. Stücklistenauflösung aufgrund von Gozintographen
2. Auftragsterminierung in der Netzplantechnik
3. SQL-Abfragen anhand von Datenbanken
4. Lösen von Interpolationsgleichungen für die Geo-Interpolation
5. Bestimmen des kürzesten Pfades

(Christ, et al., 2022)

3.3 Benutzergruppen in der Lernplattform ALADIN

Lehrende:

Die Lehrenden definieren zusammen mit den Entwicklern die Inhalte und Struktur der Aufgaben sowie die bereitzustellenden Lösungshilfen. Sie geben fachlichen Input, welche Modellierungsaufgaben abgedeckt werden sollen und liefern dazu Referenzlösungen. Die Lehrenden nutzen die Lernplattform zur Erstellung individueller Prüfungsaufgaben.

Studierende:

Die Studierenden lösen die von der Lernplattform automatisch generierten Aufgaben aus verschiedenen Themenbereichen, um ihr Wissen anzuwenden und zu vertiefen. Dabei können sie die Komplexität der Aufgaben an ihren individuellen Kenntnisstand anpassen, was zu einer individuellen und anregenden Lernerfahrung führt. Die Studierenden erhalten bei Bedarf schrittweise Lösungshilfen und sofortiges Feedback zur Korrektheit ihrer Lösungen. Sie können ihre Lösungsversuche aufzeichnen und mit Kommilitonen sowie Lehrenden teilen.

Entwickler:

Die Entwickler setzen die fachlichen Vorgaben technisch um. Sie programmieren die Generatoren für die parametrisierbare, zufallsbasierte Erzeugung neuer Aufgaben und integrieren die Lösungshilfen und Korrekturfunktionalität. Die Entwickler stellen die Lernplattform über das Internet bereit und sorgen für Benutzerfreundlichkeit. Sie erweitern die Lernplattform um neue Aufgabentypen und passen sie an veränderte Anforderungen an.

3.4 Die Software-Architektur der Lernplattform ALADIN

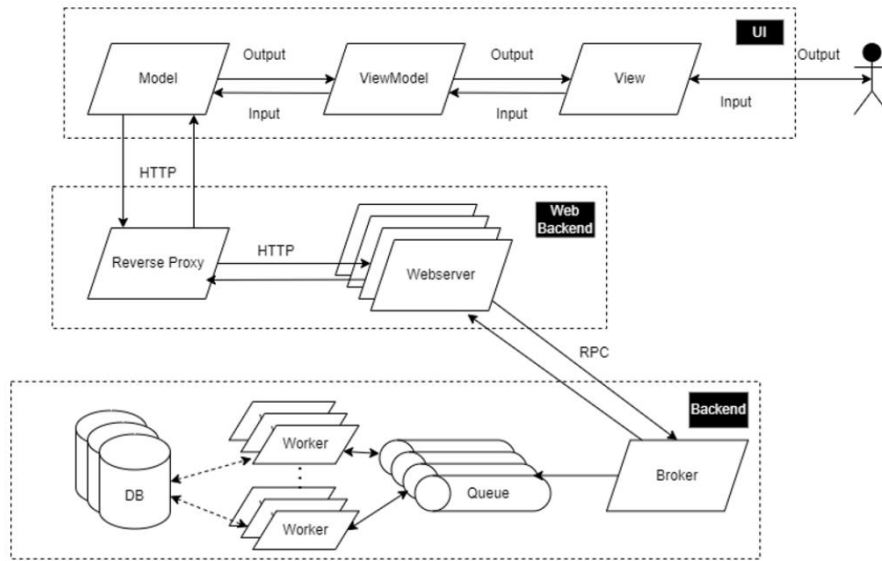


Abbildung 3: Technische Realisierung von ALADIN (Christ, et al., 2022)

3.4.1 ALADIN-Frontend

Das Frontend von ALADIN, ist als Single-Page-Webanwendung auf Basis von Vue.js umgesetzt. Abbildung 3 stellt die Struktur des Frontend dar, mit diesem Teil interagiert der Benutzer. Es folgt dem MVVM-Entwurfsmuster (Model View ViewModel), was die Umsetzung von Event Sourcing zur Aufzeichnung und Wiedergabe von Nutzerinteraktionen vereinfacht. Das Frontend wird durch einen Nginx-Webserver bereitgestellt (Christ, et al., 2022).

3.4.2 ALADIN-Backend

Das Backend von ALADIN ist als ereignisgesteuerte Microservice-Architektur auf Basis von Node.js realisiert. Es verwendet einen Event-Broker und Queues, um eingehende Events an mehrere Worker verteilen zu können. Dadurch ist das Backend dynamisch skalierbar und kann eine hohe Anzahl paralleler Anfragen verarbeiten. Zur persistenten Datenspeicherung kommen eine MongoDB und PostgreSQL-Datenbank zum Einsatz.

Die Kommunikation zwischen Front- und Backend erfolgt über ein API-Gateway, das die Anfragen auf die verschiedenen Node.js-Webserver verteilt. Diese kommunizieren untereinander über Remote Procedure Calls (RPC) und lösen entsprechende Events im

Backend aus. Jeder Aufgabentyp besitzt eine eigene Queue, sodass bei Bedarf die Anzahl der Worker für stark frequentierte Aufgabentypen erhöht werden kann.

Durch diesen schlanken und skalierbaren Aufbau als Microservice-Architektur ist ALADIN in der Lage, eine große Anzahl von parallelen Anfragen zur Aufgabenerstellung und -bearbeitung performant zu bedienen (Christ, et al., 2022).

4 Analyse der Anforderungen an Gamification in der Lernplattform ALADIN

In den folgenden Abschnitten werden die hypothetischen Anforderungen im Hinblick auf die jeweiligen Benutzergruppen beleuchtet. Auch wenn keine speziellen Umfragen durchgeführt wurden, wurden diese Annahmen aus den Bedürfnissen der Benutzer abgeleitet. Es ist zu beachten, dass zukünftige Umfragen und Untersuchungen notwendig sein könnten, um die tatsächlichen Anforderungen präziser zu ermitteln.

4.1 Anforderungen der Lehrenden an Gamification in der Lernplattform ALADIN

- Einfache und flexible Gestaltung der Gamification-Elemente
- Einfache Konfiguration der Gamification-Elemente, wie Punkte und Level

In einer zukünftigen ALADIN-Version wird ein Autorentool für Lehrende integriert, um Aufgabentypen zu erstellen. Gamification-Anforderungen für dieses Tool sind:

- Belohnung für die Erstellung vieler verschiedener Aufgaben/Lösungswege
- Belohnung für die Erstellung verschiedener Aufgabentypen

4.2 Anforderungen der Studierenden an Gamification in der Lernplattform ALADIN

- Belohnung für das Lösen verschiedener Aufgaben durch Punkte, Level
- Klares Feedback während des Lösens
- Balance zwischen Herausforderung und Fähigkeit
- Ansprechende visuelle Gestaltung der Gamification-Elemente

4.3 Anforderungen der (Weiter-)Entwickler an Gamification in der Lernplattform ALADIN

- Einfache Integration von Gamification-Elementen in die Softwarearchitektur
- Konfiguration und Parametrisierung der Gamification über JSON-Dateien
- Trennung von fachlicher Aufgabenlogik und Gamification
- Erweiterbarkeit um neue Aufgabentypen und Gamification-Elemente

5 Aktueller Stand der Technik in Bezug auf Gamification und ALADIN

5.1 Allgemeine Herangehensweisen bei der Entwicklung von Gamification

Herangehensweise in der Hochschulbildung

Die Herangehensweise bei der Entwicklung von Gamification in der Hochschulbildung basiert auf verschiedenen Designprinzipien, die entscheidend für den Erfolg sind (Tolks & Sailer, 2021). Diese Prinzipien umfassen unter anderem die Kombination von Game Design Elementen, um verschiedene Motivationsfaktoren anzusprechen. Dies führt zu aktivierendem Lernen, bei dem Studierende aktiv in den Lernprozess eingebunden werden, zum Beispiel durch Quizprogramme und Wettbewerbe. Zudem ist die Anpassung an die Zielgruppe entscheidend, um die Gamification-Elemente auf die jeweiligen Spielertypen und die Zielgruppe zuzuschneiden. Die vollständige Liste der Designprinzipien wird in Tabelle A11 im Anhang dargestellt. Die Anwendung dieser Designprinzipien soll bei der Entwicklung der Gamification in ALADIN helfen.

Herangehensweise im Unternehmen

Um den Vorgang der Gamification aus einer anderen Perspektive zu beleuchten, folgt Abbildung 4 zur Einführung und Gestaltung von Gamification im Unternehmen anhand der Forschung von Stieglitz (Stieglitz, 2015).

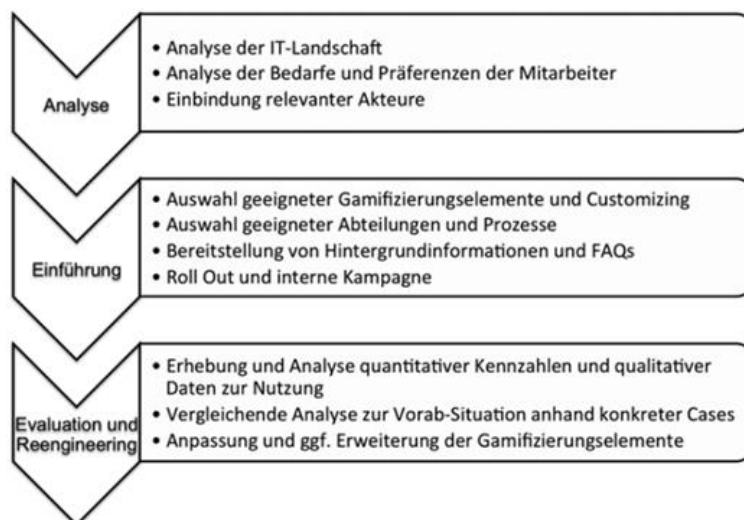


Abbildung 4: Schritte bei der Einführung von Gamification (Stieglitz, 2015)

In Abbildung 4 ist zu erkennen, dass die Einführung von Gamification in Unternehmen systematisch in drei Phasen erfolgen sollte:

Die Analyse der Ausgangslage und der Mitarbeiterbedarfe ist entscheidend für die erfolgreiche Einführung von Gamification-Elementen. Sie erfordert eine technische Infrastrukturanalyse, die Identifizierung von Mitarbeiterpräferenzen und die Berücksichtigung von Effekten und Nebenwirkungen. Die Einbindung interner Stakeholder wie Betriebsrat und Abteilungsleiter ist ebenfalls unerlässlich. Diese umfassende Analyse legt den Grundstein für eine effektive und akzeptierte Gamification-Strategie in der Organisation.

Die Vorbereitung der Einführungsphase erfordert eine gründliche Evaluierung der geeigneten Gamifizierungsansätze und deren Anpassung an die spezifischen Bedürfnisse des Unternehmens. Dabei können Praxisbeispiele als Inspiration dienen. Das Design der Gamification-Funktionen sollte intuitiv, benutzerfreundlich und transparent sein, um kognitive Überlastung und Informationsdefizite zu vermeiden. Entscheidend ist die Kopplung von Gamification-Elementen an passende Unternehmensprozesse, sei es auf allgemeiner oder spezialisierter Ebene. Klare Kommunikation über die zugrunde liegenden Mechaniken und die transparente Einführung von Gamification sind unerlässlich, um die Motivation der Mitarbeiter zu erhalten und eine Ablehnungshaltung zu verhindern. Der Rollout von Gamification sollte strategisch geplant werden, entweder bei der Einführung neuer Systeme oder bei der Erweiterung bestehender Plattformen, um von Anfang an eine spielerische Komponente zu integrieren und die Mitarbeiter aktiv einzubeziehen.

Die Evaluation und das Reengineering der Gamification-Elemente sollten kontinuierliche Prozesse sein. Durch eine laufende Evaluation lassen sich die konkreten Auswirkungen überprüfen und darauf basierend Anpassungen vornehmen. Langfristig muss die Attraktivität der Gamification-Elemente erhalten bleiben, gegebenenfalls ist ein Wechsel der Elemente ratsam.

Das Einführungsmodell für Gamification in Unternehmen in drei Phasen - Analyse, Einführung und Evaluation - ist auf die Integration von Gamification in ALADIN übertragbar. Die Analyse umfasst die Bedarfsanalyse und technische

Infrastrukturanalyse, die Vorbereitung beinhaltet Anpassung und intuitive Gestaltung, sowie transparente Kommunikation, und die kontinuierliche Evaluation sichert die langfristige Attraktivität der Gamification-Elemente in ALADIN.

5.2 Technischer Zustand des ALADIN-Frameworks

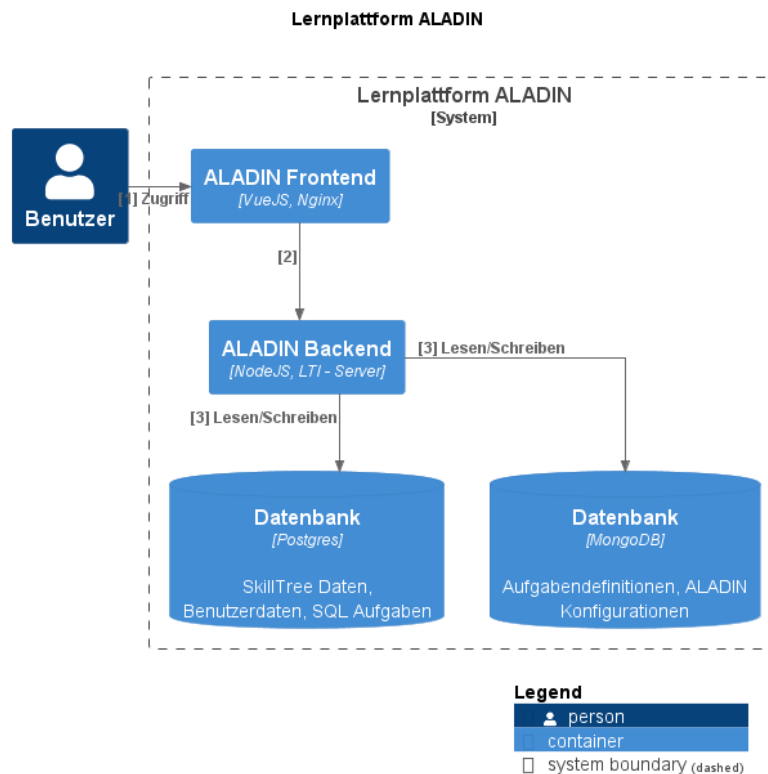


Abbildung 5: Aktueller Stand des ALADIN-Frameworks (Quelle: Eigene Darstellung)

Die Lernplattform ALADIN bietet umfangreiche Möglichkeiten zur Integration von Gamification. In der Abbildung 5 ist die aktuelle Architektur des ALADIN-Frameworks dargestellt. Hierbei sind sowohl das Backend als auch das Frontend als entscheidende Komponenten hervorzuheben. Diese beiden Teile der Plattform bieten Raum für Erweiterungen und Möglichkeiten zur Gamifizierung.

Das Frontend, ermöglicht die Integration von Elementen, die direkt von den Benutzern verwendet werden können. Hierzu gehören unter anderem Funktionen zur Verbesserung des Nutzererlebnisses und der Interaktion mit der Plattform. Auch können hier die Aktionen der Benutzer abgefangen und gespeichert werden.

Das Backend hingegen bietet die Grundlage für tiefgreifendere Gamification-

Integrationen. Hierbei können verschiedene Aspekte betrachtet werden, wie Berechnungen für verschiedene Gamification-Elemente, welche im Frontend verwendet werden. Zusätzlich soll geprüft werden, welche Benutzerdaten durch die Gamification bereitgestellt werden müssen.

Die folgenden Abschnitte erläutern technische Bestandteile, die für die Integration der Gamification essenziell sind.

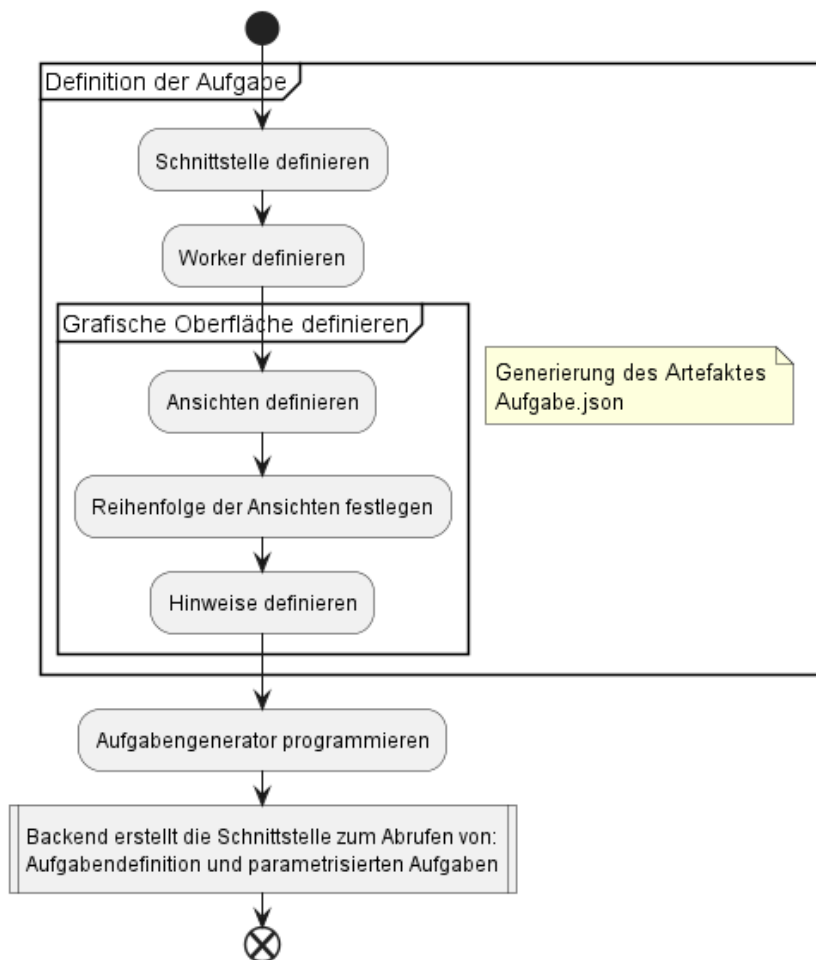


Abbildung 6: Prozess der Aufgabendefinition durch Lehrende und Entwickler in ALADIN (Quelle: Eigene Darstellung)

Abbildung 6 veranschaulicht den Prozess der Aufgabendefinition in ALADIN, dieser wird jedes Mal durchlaufen, wenn neue Aufgabentypen erstellt werden. Im Folgenden wird nun konkreter auf Details und Besonderheiten eingegangen.

Das Herzstück des ALADIN-Frameworks sind seine Aufgabengeneratoren. Nur mit diesen können am Ende parametrisierte Aufgaben generiert und den Studierenden zum

Lösen bereitgestellt werden. Ein weiterer wichtiger Bestandteil ist die Aufgabe.json. In dieser wird die Struktur der Aufgabe festgehalten und mit dem Aufgabengenerator verknüpft. So besteht eine Aufgabe immer aus mehreren Oberflächen. Für jede Oberfläche werden die entsprechenden Elemente, z.B. Textfelder, Matrizen etc., definiert. Elementen können (Hilfs-)Funktionen zugeordnet werden, z.B. „Kopieren“, „Ergänze Nullen“, welche dann vom Benutzer verwendet werden können. So gibt es auch Teillösungen und Komplettlösungen, welche sich der Nutzer ausgeben lassen kann. Entscheidend ist, dass in der Aufgabe.json alle Elemente der Oberfläche eindeutig gekennzeichnet werden. Zur Laufzeit lassen sich dann die Zustände der Elemente abfragen.

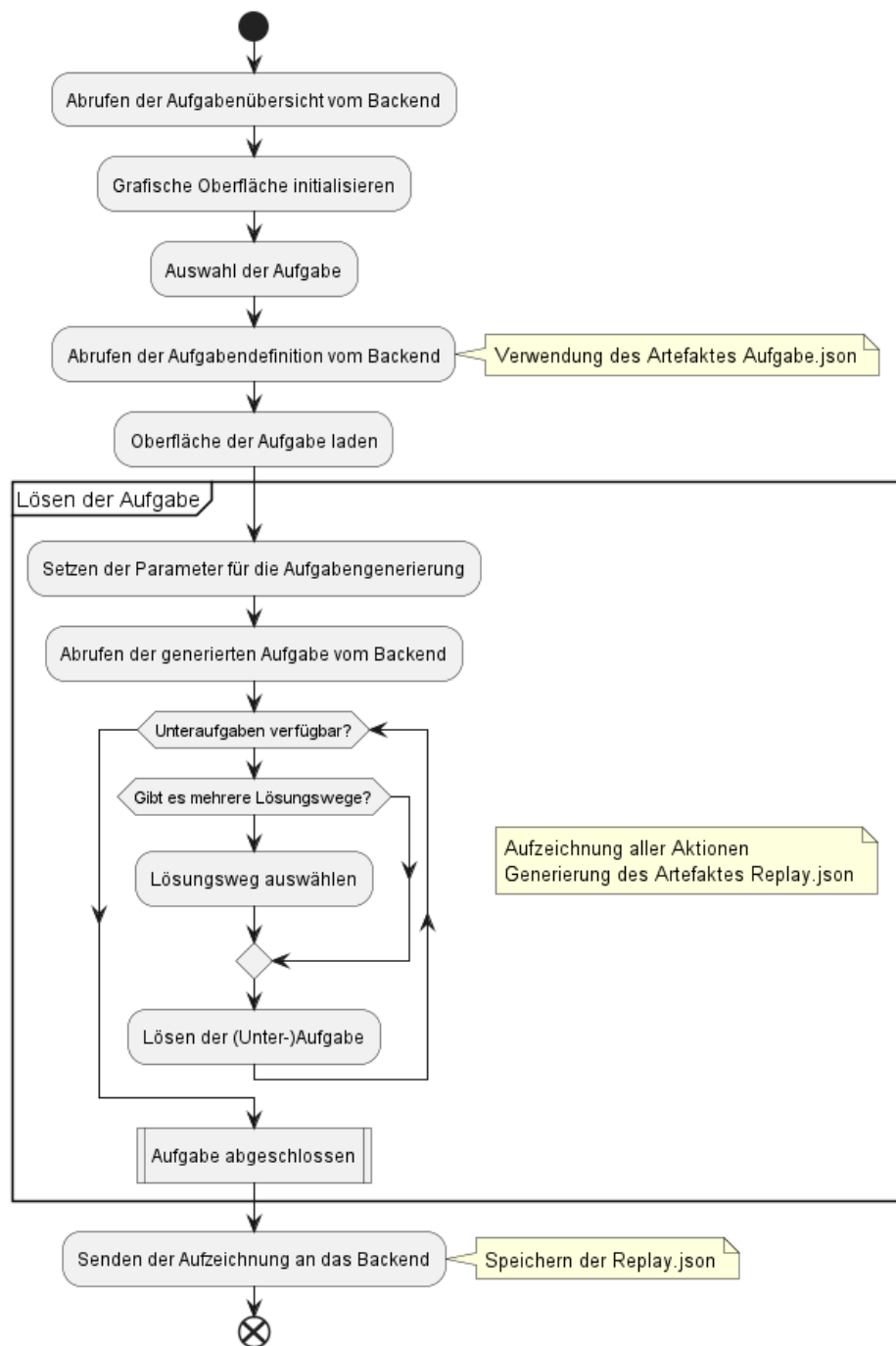


Abbildung 7: Aufgabeninitialisierung und -bearbeitung durch Studierende in ALADIN (Quelle: Eigene Darstellung)

Die Aufgabeninitialisierung und -bearbeitung durch Studierende in ALADIN wird durch Abbildung 7 verdeutlicht. So werden in den ersten Schritten die Aufgabenübersicht und die gewählte Aufgabe vom Backend aufgerufen und zugehörige Oberflächen angezeigt. Danach können die Studierenden die ausgewählte Aufgabe lösen. Während der Aufgabenbearbeitung wird im Frontend eine Aufzeichnung erstellt, die Grundlage dieser

ist die Aufgabe.json. Anhand der Aufgabe.json werden Pfade mit zugehörigen Werten und Zeitstempeln generiert. So beinhaltet die Aufzeichnung alle Aktionen des Benutzers, inklusive Informationen über Zeiten, eingetragene Werte, verwendete Hilfsmethoden. Diese Aufzeichnung kann jederzeit durch den Benutzer abgespeichert werden, dann wird sie an das Backend gesendet und dort persistiert.

5.3 Gamification-Frameworks und Gamification-Plattformen

Tabelle 4: Vergleich ausgewählter Gamification Lösungen (Quelle: Eigene Tabelle)

Lösung	Plattform	Letztes Release	Beschreibung
Gamification Modules for Angular (https://www.npmjs.com/package/gamification)	Web (Angular)	vor 5 Jahren	- Webservice (wenige Informationen)
Gamification UX Package (https://www.npmjs.com/package/@ziqni-tech/gamification-ux-package)	Frontend (JS)	vor 5 Monaten	- Widgets zum Einsatz in Webapplikationen
Moodle Level UP XP – Gamification (https://moodle.org/plugins/block_xp)	Moodle	vor 3 Monaten	- Speziell zum Gamifizieren von Moodle Seiten
Django-Gamification (https://pypi.org/project/django-gamification/#history)	Web (Django)	vor 5 Jahren	- Einfaches Badge-System Prototypisches Frontend und Backend
SkillTree Plattform (https://skilltreeplatform.dev/overview/)	Frontend (JS, Vue.js, React.js, Angular), Backend (Java Spring Boot)	Aktuell	- Flexible Open-Source Mikro-Learning-Gamification-Plattform - Kann als eigenständige Trainingsplattform oder in bestehende Webanwendungen integriert werden

Tabelle 4 bietet einen Überblick über aktuelle Lösungen zum Zeitpunkt der Analyse. Es fällt auf, dass einige dieser Lösungen nicht mehr aktiv weiterentwickelt werden und bereits veraltet sind. Oftmals fehlen umfassende Informationen zu diesen Projekten, da viele von Einzelpersonen entwickelt wurden. Besonders vielversprechend erscheinen jedoch das Gamification UX Package und die SkillTree Plattform. Da die SkillTree Plattform die reifste und umfassendste Lösung ist, wird diese für den weiteren Einsatz in ALADIN in Betracht gezogen.

5.4 Analyse der SkillTree Plattform zur Gamification der Lernplattform ALADIN

Übersicht zu SkillTree

SkillTree ist eine Mikro-Lern-Gamification-Plattform, die verschiedene Integrationsmöglichkeiten unterstützt, um Gamification in bestehende Anwendungen zu integrieren. Die Plattform ermöglicht es, Schulungen auf verschiedene Arten durchzuführen:

1. Über das SkillTree-Dashboard in Learning Management System (LMS)-Manier:
Hier können Schulungen direkt verwaltet werden.
2. Integration in bestehende oder neue webbasierte Anwendungen: SkillTree kann in diese Anwendungen integriert werden, wobei die Fähigkeitanzeige von SkillTree mit den Client-Bibliotheken direkt eingebettet wird.

SkillTree bietet eine Lösung, um Gamification günstig und zeitsparend umzusetzen. Zur Integration von Gamification in bestehende Anwendungen, werden Display-Bibliotheken für eine interaktive Nutzerrangliste und Fortschrittsanzeige bereitgestellt. Außerdem ist Beispiel-Code für die Integration der Bibliotheken und weitere Services durch SkillTree verfügbar.

SkillTree ist kostenlos und Open Source und unter der Apache 2-Lizenz veröffentlicht.

Features des Frameworks

- Erstellung von Trainingsprofilen (Projekten)
- Aufstieg in Stufen (Level) innerhalb eines Projekts
- Gruppierung von Fähigkeiten unter Themengebiete
- Vergabe von Abzeichen an Benutzer
- Selbstberichterstattung der Benutzer mit verschiedenen Genehmigungsmechanismen
- Export von Fähigkeiten in den Skills Katalog für die Wiederverwendung

- Wissenstests mit Quizzen für den Erwerb von Fähigkeiten und Punkten
- Durchführung von Umfragen zur Rückmeldung und Datensammlung
- Festlegen von Lernpfaden für die Reihenfolge des Fähigkeitserwerbs
- Metriken zur Sichtbarkeit von Fachwissen und Wissenslücken auf Unternehmensebene
- Reports über fehlerhaftes Verhalten der Clients, z.B. melden nicht vorhandener Skills

Erläuterung zur Verwendung der SkillTree Plattform

Um SkillTree zu verwenden, muss ein Projekt angelegt werden. Dieses Projekt beinhaltet Themengebiete, welche wiederum Fähigkeiten (Skills) gruppieren. Skills sind der wichtigste Baustein der Plattform. Punkte sind eng mit Skills verknüpft: Durch Melden von Skills werden Punkte vergeben. Mit gesammelten Punkten steigen Anwender*innen in den vordefinierten Levels auf. Die Skill-Definition umfasst, wie oft Skills gemeldet werden müssen, um als vollständig erlernt zu gelten. Auch die Vergabe von Abzeichen (Badges) ist von Skills abhängig. Beispielsweise kann ein Abzeichen erworben werden, indem drei verschiedene Skills vollständig abgeschlossen wurden. Mehrere Skills können eine Skill-Gruppe (Fähigkeitsgruppe) bilden. Diese repräsentiert einen separaten Skill. Die Skill-Gruppe gilt als abgeschlossen, sobald alle Skills in ihr erworben wurden. In manchen Fällen genügt auch der Erwerb einer festgelegten Anzahl von Skills innerhalb der Gruppe.

Beispiel zu einer Skill-Gruppe:

Die „Stücklistenauflösung“ stellt ein Themengebiet in der SkillTree-Plattform dar. Eine Skill-Gruppe könnte ein Lösungsweg, beispielsweise „Matrizenmultiplikation“, sein. Die Skill-Gruppe fasst mehrere spezifische Fähigkeiten wie „Adjazenzmatrix ablesen“, „Direktbedarfsmatrix multiplizieren“, „Gesamtbedarf bestimmen“ und „Sekundärbedarf bestimmen“ zusammen. Diese Skill-Gruppe ist abgeschlossen, wenn der Lernende alle diese spezifischen Skills beherrscht.

Softwarearchitektur der Plattform

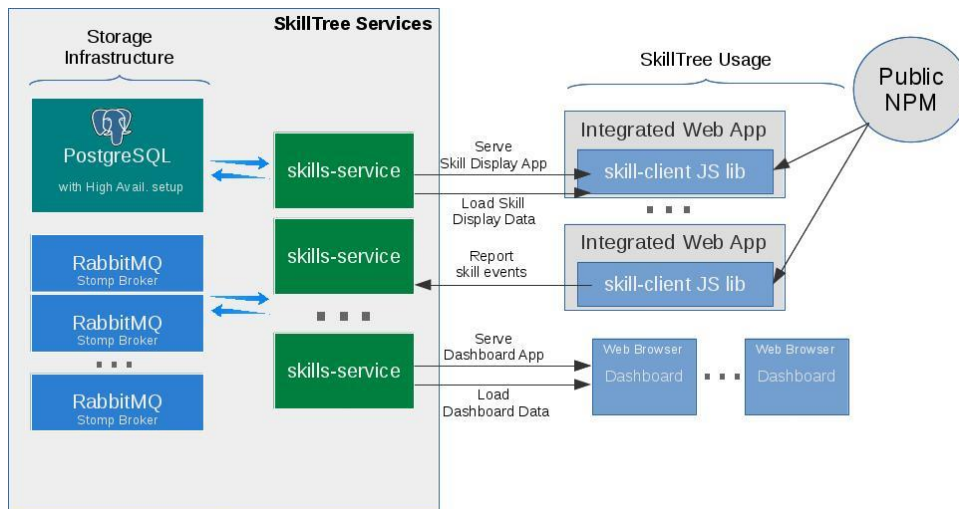


Abbildung 8: Softwarearchitektur eines produktiven SkillTree Setups
(<https://skilltreeplatform.dev/assets/img/SkillsServiceArchitecture.89964c54.jpg>)

Um den Service zu starten, wird lediglich eine PostgreSQL-Datenbank benötigt. Eine Produktionsinstallation des SkillTree Skills-Service erfordert jedoch die in Abbildung 8 dargestellte Infrastruktur:

1. PostgreSQL: Dieser relationale Speicher wird für Projektdefinitionen und Skill-Events verwendet.
2. RabbitMQ Stomp-Broker: Dieser Broker wird benötigt, um die WebSocket-Funktionalität zu unterstützen.

Der SkillTree Skills-Service kann so konfiguriert werden, dass er PostgreSQL und Stomp-Broker verwendet.

Die integrierten Web-Clients verwenden Skill-Client-Bibliotheken wie Vue.js und React.js, um über iFrames auf die Skills-Display-App und Daten vom Skills-Service zuzugreifen. Diese Bibliotheken dienen als schlanke Wrapper und abstrahieren technische Details für die Benutzer. Sie ermöglichen es auch, Skill-Ereignisse über den „Report Skill Event Endpoint“ zu melden und Benachrichtigungen über diese Ereignisse zu erhalten. Dies bietet Flexibilität für verschiedene Integrationsanforderungen und ermöglicht die nahtlose Einbindung von Skill-Ereignissen über WebSockets.

Der SkillTree Skills-Service besitzt selbst ein Frontend und kann somit als eigenständige

Anwendung verwendet werden. In dieser können alle Features der Plattform, wie das Anpassen der Skill-Strukturen, genutzt werden.

Limitationen der SkillTree Plattform

1. Ein einzelner Webclient kann immer nur ein Trainingsprofil (Projekt) darstellen.
2. Die Punktzahl ist direkt an die Skills gebunden und kann nicht separat angepasst werden.
3. Die Verschachtelung von Themengebieten in der Struktur „Projekt – Themengebiet – Skill-Gruppe – Skill“ wird nicht unterstützt.
4. Frontend-Komponenten lassen sich zwar konfigurieren, aber nicht direkt bearbeiten, da diese durch das Backend vollständig ausgeliefert werden.
5. Level-Anzeige des Clients zeigt immer das gesamte Level für das Projekt

Fazit zur SkillTree Plattform

SkillTree bietet mit seinen Skill-Client-Bibliotheken eine nahtlose Integration von Gamification in ALADIN. Die Plattform erfüllt viele der in Abschnitt 4 analysierten Anforderungen, welche durch die Konfiguration von Badges, Levels und Skills umgesetzt werden können. Dies ermöglicht eine flexible Gestaltung der Gamification-Elemente in ALADIN. Durch die Nutzung von SkillTree können Lehrende und (Weiter-)Entwickler Gamification-Elemente in ALADIN einfach integrieren und anpassen. Dies bietet die Möglichkeit, ein interaktives und motivierendes Lernerlebnis zu schaffen, das die Bedürfnisse der Studierenden und die Anforderungen der Lehrenden erfüllt.

6 Analyse der Integrationspunkte und relevanter Gamification-Elemente für die Lernplattform ALADIN

6.1 Analyse von Integrationspunkten für die inhaltliche Integration der Gamification in ALADIN

Um Gamification-Elemente zu integrieren, werden zunächst inhaltliche Integrationsmöglichkeiten überprüft.

Aufgabenparametrisierung durch Studierende:

Ein wichtiger Ansatz besteht darin, Studierenden die Möglichkeit zu geben, den Schwierigkeitsgrad der Aufgaben selbständig an ihr eigenes Verständnis anzupassen. Bei diesem Schritt kann schon eine geringe Belohnung vergeben werden, um die Motivation zu steigern.

Aufgabentypen und Schwierigkeitsgrade:

Die Integration von Gamification sollte an die Art der Aufgabe und deren Schwierigkeitsgrad angepasst sein. Zum Beispiel könnten schwierigere und zeitaufwändigere Aufgaben mehr Belohnungen bieten als einfachere und kürzere Aufgaben. Zudem gibt es auch Aufgaben, die zwar nicht schwierig sind, bei denen aber das Risiko für Flüchtigkeitsfehler höher ist, dieser Einfluss sollte separat betrachtet werden, beispielsweise beim Ablesen der Adjazenzmatrix anhand des Gozintographen.

Anzahl gelöster Aufgaben:

Anhand aller gespeicherten Aufzeichnungen kann die Anzahl bestimmt werden, wie oft Aufgabentypen gelöst wurden, auch eine Gruppierung nach der Schwierigkeit ist möglich. Mithilfe dieser Informationen können Auszeichnungen vergeben und Ranglisten erstellt werden.

Aufgabenbearbeitung:

Die eigentliche Integration von Gamification erfolgt in Aufgaben und Übungen. Auch hier können Punkte, Belohnungen oder Badges für das erfolgreiche Lösen von Aufgaben vergeben werden. Dies motiviert die Studierenden, sich stärker auf die Lerninhalte zu konzentrieren. Besonders bei mehrstufigen Aufgaben ist die Vergabe von Punkten sinnvoll, da dies die Studierenden dazu anregt, die gesamte Aufgabe zu lösen.

Die Kombination dieser Integrationspunkte kann dazu beitragen, ein motivierendes und anpassbares Gamification-Erlebnis in ALADIN zu schaffen, um die Studierenden zur aktiven Teilnahme am Lernprozess zu ermutigen.

6.2 Analyse von Integrationspunkten für die technische Integration der Gamification in ALADIN

Um Gamification-Elemente zu integrieren, werden nun technische Integrationsmöglichkeiten geprüft.

Wie in Kapitel 5.2 beschrieben wird, jede Aktion in der Lernplattform aufgezeichnet. Aktuell wird diese Aufzeichnung verwendet, um eine Aufgabe im Nachhinein erneut abspielen zu können. Somit könnte diese Aufzeichnung als Basis für die Berechnung von Punktzahlen dienen. Die Aufzeichnung ist sowohl permanent im Frontend abfragbar als auch stapelweise im Backend, sobald die Aufzeichnung vom Frontend ans Backend übertragen wurde. Im Folgenden wird die Aufzeichnung als Integrationspunkt betrachtet und geprüft, ob die Integration im Frontend oder Backend erfolgen sollte.

In Tabelle 5 im Anhang, wird anhand verschiedener Aspekte ein Vergleich zwischen der Integration im Frontend und Backend durchgeführt. Die Aspekte anhand der Anstriche (+) für positive oder (-) für negative Eigenschaften aufgelistet.

Tabelle 5: Vergleich zwischen der Integration im Frontend und Backend (Quelle: Eigene Tabelle)

Aspekt	Integration im Frontend	Integration im Backend
Entkopplung von der Client-Logik	<ul style="list-style-type: none"> - Die Client-Logik ist eng mit den Gamification-Elementen verknüpft, was die Flexibilität einschränken kann. - Änderungen erfordern oft Updates der Client-App. 	<ul style="list-style-type: none"> + Gute Entkopplung von der Logik des Clients. + Änderungen können im Backend ohne Änderungen auf dem Client umgesetzt werden.

Berechnungen	- Die Berechnungen erfolgen auf dem Gerät des Benutzers, was die Leistung beeinflussen kann.	+ Die Berechnungen laufen unabhängig von der Leistung des Benutzergeräts, was die Konsistenz gewährleistet. - Aufwändige Berechnungen könnten sich im Backend stauen
Echtzeit-Feedback	+ Direktes Echtzeit-Feedback für Benutzer ist möglich.	- Echtzeit-Feedback im Backend ist schwieriger zu realisieren. - Feedback kann verzögert sein, abhängig von der Synchronisierungsfrequenz der Aufzeichnung. - WebSockets sind erforderlich, um Echtzeitaktualisierungen zu ermöglichen.
Zugriff auf Nutzerdaten	+ Volle Zugriff auf alle Nutzerdaten und -interaktionen im Frontend.	- Begrenzter Zugriff auf Nutzerdaten im Backend. - Datenübertragungen zwischen Frontend und Backend können aufwändig sein.

6.3 Identifizierung relevanter Gamification-Elemente für ALADIN

Anhand der Gamification-Elemente aus Tabelle 2 und den Anforderungen aus Kapitel 4 können folgende Elemente als relevant zur Gamifizierung von ALADIN betrachtet werden:

Punktesystem, Level:

- Punkte können beim Bearbeiten von Aufgaben vergeben werden.
- Variable Punktzahl: je nach Schwierigkeit oder Komplexität kann sich die

Punktzahl unterscheiden, was sehr viel Flexibilität bietet.

Herausforderungen/Quests:

- Prüfungsvorleistungen könnten als Main-Quest und Hausaufgaben als Side-Quest betrachtet werden. Somit könnte ALADIN mehr Aspekte der Hochschule abdecken.

Serienbelohnung:

- Das wiederholte Lösen von Aufgaben kann durch zusätzliche Punkte belohnt werden.

Badges/Auszeichnungen:

- Die Vergabe von Badges oder Auszeichnungen, kann anhand von der Anzahl an absolvierten Aufgaben geschehen.
- Es könnte auch zeitlich begrenzte Abzeichen geben, die beim Einführen von neuen Aufgabentypen vergeben werden können.

Ranglisten:

- Es können Punkte in verschiedenen Kategorien/Aufgabentypen gesammelt und als Rangliste dargestellt werden.

Fortschrittsbalken:

- Fortschrittsbalken können anhand der Anzahl gelöster Aufgaben oder Punkte generiert und dargestellt werden.

6.4 Bewertungsmatrix der Integrationspunkte und den relevanten Gamification-Elementen in ALADIN

Tabelle 6: Bewertung der inhaltlichen Integrationspunkte und den relevanten Gamification-Elementen in ALADIN (Quelle: Eigene Darstellung)

Gamification- Elemente Integrations- punkte	Punktesystem, Level	Heraus- forderungen / Quests	Serien- belohnung	Badges / Auszeich- nungen	Ranglisten	Fortschritts- balken
Aufgaben- parametrisierung	✓	×	×	×	×	×
Unterschiedliche Aufgabentypen	✓	✓	×	✓	✓	×
Verschiedene Schwierigkeitsgrade	✓	✓	×	✓	✓	×
Anzahl gelöster Aufgaben	×	✓	✓	✓	✓	✓
Aufgabenbearbeitung	✓	✓	✓	×	×	✓
Auswertung	sehr gute inhaltliche Integration möglich	sehr gute inhaltliche Integration möglich	mäßige inhaltliche Integration möglich	gute inhaltliche Integration möglich	gute inhaltliche Integration möglich	mäßige inhaltliche Integration möglich

Legende: geeignet = ✓ nicht geeignet = ×

Tabelle 6 zeigt, dass das Punktesystem und Herausforderungen/Quests besonders gut in ALADIN integriert werden können. Eine geringe Punktevergabe kann bereits bei der Parametrisierung der Aufgaben erfolgen, und bei schwierigeren Aufgaben oder verschiedenen Aufgabentypen können die Punkte angepasst werden. Für das Gamification-Element der Serie bieten sich wenige Integrationspunkte an, da die Belohnung nicht von der Schwierigkeit der Aufgaben oder dem Aufgabentypen abhängt, sondern von der Häufigkeit der Bearbeitung.

Tabelle 7: Bewertung der technischen Integrationspunkte den relevanten Gamification-Elementen in ALADIN (Quelle: Eigene Darstellung)

Gamification- Elemente Integrations- punkte	Punktesystem, Level	Heraus- forderungen / Quests	Serien- belohnung	Badges / Auszeich- nungen	Ranglisten	Fortschritts- balken
Aufzeichnung im Frontend	✓	X	X	X	X	✓
Aufzeichnung im Backend	X	✓	✓	✓	✓	✓
Auswertung	Integration ins Frontend	Integration ins Backend	Integration ins Backend	Integration ins Backend	Integration ins Backend	Integration ins Frontend und Backend

Legende: geeignet = ✓ nicht geeignet = X

Tabelle 7 bewertet die technischen Integrationspunkte für die relevanten Gamification-Elemente in ALADIN. Die Entscheidungen zur Integration hängen von verschiedenen Faktoren ab. Zum Beispiel ist die Auswertung von Punkten im Frontend effizienter, da eine Übertragung aller Daten ans Backend ineffizient wäre. Die Rangliste hingegen sollte im Backend verarbeitet werden, da hier erhebliche Rechenleistung erforderlich ist. Bei den Fortschrittsbalken kann die Integration ins Backend sinnvoll sein, um den Fortschritt anhand der Anzahl gelöster Aufgaben zu bewerten. Dies liegt daran, dass die Berechnung die Analyse mehrerer Aufzeichnungen einschließt. Wenn jedoch der Fortschritt während der Aufgabenbearbeitung angezeigt werden soll, ist die Integration ins Frontend angebracht.

7 Konzept(ion)/Entwurf der Gamification-Elemente für die Lernplattform ALADIN

7.1 Besonderheiten zur Konfiguration der Gamification-Elemente und Betrachtung von Folgefehlern

7.1.1 Konfiguration der Gamification-Elemente

Die Konfiguration der Gamification-Elemente in ALADIN stellt eine weitere Herausforderung dar. Diese Elemente müssen in der Lage sein, sich konfigurativ an verschiedene Aufgabentypen anzupassen. Diese Anpassung sollte deklarativ erfolgen und an die unterschiedlichen Anforderungen der Aufgabenstellungen gebunden sein. Dieser Ansatz ist von entscheidender Bedeutung, um sicherzustellen, dass die Gamification-Elemente flexibel und universell einsetzbar sind. Diese Flexibilität ermöglicht es, die Lernplattform kontinuierlich an die sich wandelnden Anforderungen und Bedürfnisse der Benutzer anzupassen und so eine effektive und nachhaltige Lernerfahrung zu gewährleisten.

7.1.2 Wertung von Folgefehlern

Ein weiterer bedeutender Aspekt in der Konzeption ist die Reaktion auf Folgefehler. Zukünftige Versionen von ALADIN werden die Möglichkeit bieten, Folgefehler zu akzeptieren und die Lösung trotzdem fortzusetzen. Hierbei ist jedoch zu beachten, dass Studierende bewusst Teilaufgaben falsch lösen können, um den nachfolgenden Teil der Aufgabe zu erleichtern. Ein solches Verhalten könnte beispielsweise bei der Matrixmultiplikation auftreten, wenn absichtlich eine Nullmatrix abgelesen wird, um die nachfolgenden Aufgaben zu vereinfachen. Bei dieser Situation sollten dem Studierenden keine Punkte zugeordnet werden. Die Prüfung, ob Eingaben bewusst falsch gemacht wurden, um die nachfolgenden Aufgaben zu erleichtern, liegt nicht im Umfang dieser Arbeit und wird daher nicht weiter beachtet.

7.2 Betrachtung verschiedener Ansätze zur Integration des Punktesystems und der Serienbelohnung in ALADIN

Anmerkung

Während eine Aufgabe bearbeitet wird, ist es den Studierenden möglich, in eine vorherige Oberfläche zu wechseln. In schon abgeschlossenen Oberflächen können dadurch erneut Eingaben getätigt werden. Eine umfassende Betrachtung dieses speziellen Verhaltens würde den Rahmen dieser Arbeit sprengen. Deshalb werden hier nur Ansätze mit einem linearen Verlauf der Aufgabe analysiert.

7.2.1 Generelles Vorgehen bei allen Ansätzen

Generell beinhaltet die Aufzeichnung sehr viele Daten, die für die Gamification und die Punkteberechnung nicht notwendig sind. Die Aufzeichnung wird zuerst nach relevanten Pfaden gefiltert. Somit fallen Informationen zum Zoomen oder Verschieben von Oberflächenelementen weg. Relevant sind beispielsweise Eingaben des Nutzers oder das Verwenden von Hilfsmethoden.

7.2.2 Erster Ansatz – Konvertierung der Schritte zu einem Objekt

Dieser Ansatz beschreibt eine Möglichkeit, alle Aufzeichnungsschritte stapelweise zu verarbeiten. Vorab müssen im ALADIN-Framework zusätzliche Metriken eingeführt werden, welche für die Berechnung der Punktzahl verwendet werden.

Anpassungen:

1. Definition von weiteren Metriken in der Aufgabe.json: Es muss definiert werden, wie viele Punkte pro Ansicht möglich sind und wie viel Zeit dafür verfügbar ist.
2. Definition der Auswirkung von verwendeten Hilfsmethoden an den jeweiligen Oberflächenelementen.

Verarbeitungsschritte:

1. Konvertierung aller Schritte der Aufzeichnung in ein JavaScript-Objekt - Aufzeichnungsobjekt:
 - Durchsuchen dieses Aufzeichnungsobjekts anhand der Zeitstempel der Schritte
2. Auswertung des Aufzeichnungsobjekts:
 - iterative Prüfung aller Schritte auf Korrektheit der Eingaben des Studierenden, indem die Eingaben mit der Musterlösung abgeglichen werden
3. Berechnung einer Punktzahl pro Ansicht anhand der Eingaben und zuvor definierten Metriken
4. Übertragung der berechneten Punktzahl an ein separates Backend für die Gamification

Probleme des ersten Ansatzes:

1. Komplexe Prüfung auf Korrektheit, da die Prüfung von den Oberflächenelementen abhängig ist.
2. Unnötige Konvertierung aller Schritte zu einem JavaScript-Objekt. Die Schritte können atomar betrachtet werden und somit sequenziell nacheinander verarbeitet werden.
3. Metriken stehen an unterschiedlichen Stellen. Die Metriken für die Ansichten werden an der Aufgabe.json definiert. Die Metriken zu den Auswirkungen verwendeter Methoden stehen an den jeweiligen Oberflächenelementen. Lösungsinformationen und Eingaben des Benutzers stehen im eigens konvertierten Aufzeichnungsobjekt.

7.2.3 Zweiter Ansatz – Erweiterung des Replays

Bei diesem Ansatz wird vorab der bestehende Code des ALADIN-Frameworks angepasst, um einige Probleme des ersten Ansatzes zu umgehen.

Anpassungen:

1. Erweiterung der Aufzeichnung um Validierungsdaten: Diese geben Auskunft, ob eine Nutzereingabe gültig, also gefüllt und im richtigen Datentyp, und inhaltlich korrekt ist. Diese Validierungsdaten werden auf unterschiedlichen Ebenen eingeführt. Auf Feld-Ebene geben diese Auskunft, ob die Eingabe korrekt und gültig ist. Auf Element-Ebene geben diese Auskunft, ob das gesamte Oberflächenelement korrekt und gültig ist. Auf Ansicht-Ebene geben diese Auskunft, ob alle Elemente der Oberfläche gültig und korrekt sind.
2. Anpassung der Aufgabe.json im Hinblick auf die Hilfsmethoden: Anders als beim ersten Ansatz wird die Auswirkung der Methoden nun in der Aufgabe.json festgehalten. Dafür muss die Definitionsstruktur angepasst werden. Es wird jetzt die Beschreibung der Methode und deren Auswirkung innerhalb eines Objektes definiert.

Verarbeitungsschritte:

1. Gefilterte Schritte werden sequenziell betrachtet. Durch die obigen Anpassungen ist aus der Aufzeichnung zu entnehmen, ob die Benutzereingabe korrekt war.
2. Berechnung der Punktzahl pro Ansicht anhand aller korrekten Eingaben in den Oberflächenelementen und den verwendeten Methoden.
3. Anpassung der Punktzahl an die maximale Punktzahl pro Ansicht.
4. Übertragung der Punktzahl an das Backend.

Konzeption der Punkteberechnung

S = Schwierigkeit der Aufgabe in den Kategorien {leicht, normal, schwer}, $S \in \{0,5; 1; 1,5\}$

Die Schwierigkeit kann anhand der Parameter der Aufgabengenerierung bestimmt

werden.

p = Anteil richtiger Eingaben im Verhältnis der Gesamteingabefelder in %

t = Dauer der Bearbeitung der Aufgabe in s

P = maximale Punktzahl für diese Ansicht (Aufgabe.json)

T = maximale Dauer für die Bearbeitung dieser Ansicht in s (Aufgabe.json)

Formel 1: Punkteberechnung

$$\text{Punktzahl} = p * P * \frac{T}{t} * S$$

Erläuterung zur Punkteberechnung:

Die Punkte orientieren sich an den maximalen Prüfungspunkten, damit die Studierenden während des Bearbeitens ein Gefühl dafür bekommen, wie gut diese in einer realen Aufgabe abgeschnitten hätten.

Wird weniger als die maximale Zeit benötigt, dann wird die Punktzahl mit einem Faktor > 1 multipliziert und somit erhöht. Wird mehr als die maximale Zeit benötigt, dann werden Punkte abgezogen, indem der Faktor < 1 wird.

Um schwierige Aufgaben zu belohnen, kann $S > 1$ sein. Damit sind mehr (Gamification-) Punkte möglich als maximal definiert.

Problem des zweiten Ansatzes:

Bei diesem Ansatz sind mehrere Anpassungen am ALADIN-Framework notwendig.

7.2.4 Dritter Ansatz – Vergabe definierter Punktzahl pro Pfad

Für diesen Ansatz werden alle Anpassungen am ALADIN-Framework vom zweiten Ansatz als gegeben betrachtet.

Anpassungen:

Für diesen Ansatz wird eine separate Punktedefinition angelegt. In dieser wird eine feste Punktzahl bestimmten Pfaden zugeordnet. Wird durch bestimmte Benutzeraktionen nun

ein definierter Pfad ausgelöst, so erhält der Benutzer die definierte Punktzahl.

Zur Verdeutlichung ein Beispiel:

Der Pfad „nodes__6__components__0__isCorrect“ gibt an, dass das Oberflächenelement 0 in der Ansicht 6 korrekt ausgefüllt wurde, wenn dessen Wert „true“ ist. Für diesen Pfad wird definiert, dass der Benutzer 15 Punkte erhält.

Verarbeitungsschritte:

1. Gefilterte Schritte werden sequenziell betrachtet.
2. Wird ein definierter Pfad mit dem entsprechenden Wert ausgelöst, werden dafür die definierten Punkte vergeben.
3. Übertragung der Punktzahl an das Backend.

Probleme:

1. Anpassungen am ALADIN-Framework notwendig
2. Feste Definition der Punkte

7.2.5 Vergleich der Ansätze

Tabelle 8: Vergleich der Ansätze zur Integration der Gamification-Elemente in ALADIN (Quelle: Eigene Darstellung)

Kriterium	Erster Ansatz	Zweiter Ansatz	Dritter Ansatz
Punkteberechnung	individuelle Berechnung der Punktzahl		statische Vergabe der Punktzahl
Punktevergabe	Ansicht-Ebene	Ansicht-Ebene	Ansicht-Ebene, Element-Ebene, Feld-Ebene
Implementierungsaufwand	sehr hoch	hoch	gering
Konfigurationsmöglichkeit	schlecht	mäßig	einfach

Anhand des Vergleichs in Tabelle 8, ist erkennbar, dass der dritte Ansatz zwar ausschließlich statische Punkte zulässt, allerdings in den anderen Kriterien mehr

Flexibilität, weniger Aufwand und mehr Konfigurationsmöglichkeiten bietet.

7.3 Integrationskonzept unter Verwendung der SkillTree Plattform

Das folgende Integrationskonzept baut auf dem dritten Ansatz zur Integration der Gamification-Elemente auf. Anstatt allerdings Pfaden eine gewisse Punktzahl zuzuordnen, werden diese an Skills der SkillTree Plattform geknüpft. Dieses Konzept bietet die Grundlage für die Implementierung des Skill-Punkte-Systems und der Level, welche auf den Punkten aufbauen.

7.3.1 Entwurf der Schnittstelle zwischen ALADIN und der SkillTree Plattform

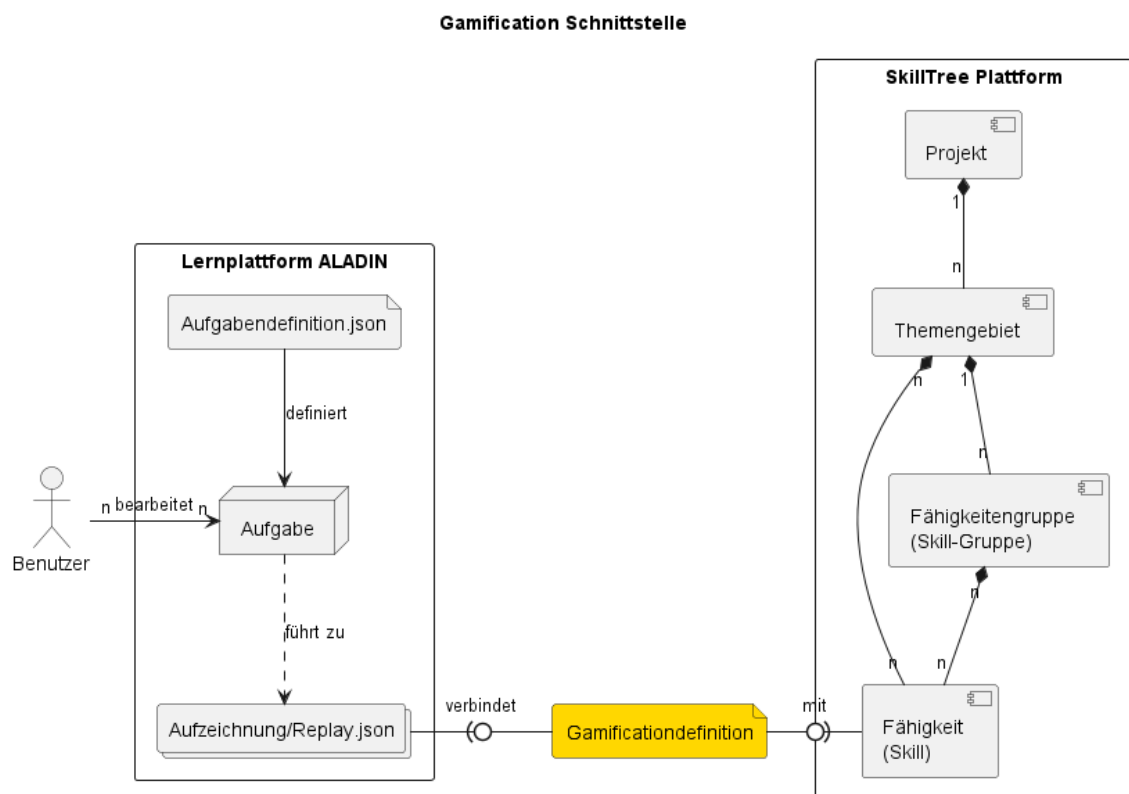


Abbildung 9: Gamificationdefinition als Schnittstelle zwischen SkillTree und ALADIN (Quelle: Eigene Darstellung)

In Abbildung 9 dargestellt, bildet die Gamificationdefinition die Verbindungseinheit zwischen ALADIN und SkillTree. Sie stellt die technische Brücke zwischen den Aufzeichnungspfaden und den Skills in der SkillTree Plattform dar. In dieser Gamificationdefinition wird pro Aufgabe festgelegt, zu welchen Pfaden welche Skills gehören.

Da ein Element sowohl die Fähigkeit zur Multiplikation von Direktbedarfsmatrizen als auch die mathematische Matrixmultiplikation darstellen kann, ist es innerhalb der Gamificationdefinition möglich, einem Pfad mehrere Skills zuzuordnen. Auf diese Weise können sowohl fachliche als auch mathematische Fähigkeiten erfasst und übermittelt werden.

Aufgaben in ALADIN sind dynamisch, so können zur Laufzeit zusätzliche Oberflächenelemente generiert werden. Ein Beispiel aus dem ALADIN Kontext ist die Stücklistenauflösung mithilfe der Matrixmultiplikation. Die Anzahl der zu multiplizierenden Matrizen ist abhängig von der Länge des Gozintographen. Daraus resultiert, dass nicht alle Elemente einen fest definierten Pfad besitzen. Um diese Besonderheit abzubilden, wird es die Möglichkeit geben, Pfade als reguläre Ausdrücke zu definieren. Dafür müssen dann zusätzliche Unterscheidungskriterien, wie z.B. dem Elementnamen, festgehalten werden, um die Elemente den entsprechenden Skills zuweisen zu können.

7.3.2 Entwurf der Skill-Struktur für die Verwendung von SkillTree mit ALADIN

Für den Aufbau der Skill-Struktur, dargestellt in Abbildung 9 rechts, gibt es mehrere Möglichkeiten. Die Vor- und Nachteile einiger betrachteter Möglichkeiten, sowie Einschränkungen durch SkillTree selbst, werden im Folgenden dargestellt.

Tabelle 9: Mögliche Skill-Strukturen für die Abbildung der Aufgaben aus ALADIN (Quelle: Eigene Darstellung)

Strukturelement	Erster Ansatz	Zweiter Ansatz	Dritter Ansatz
Projekt	Studienmodul	Studiengang	ALADIN
Themengebiet	Themengebiet	Studienmodul	
Skill-Gruppe	Lösungsweg	Themengebiet und Lösungsweg Bsp.: „Stücklistenauflösung – Matrixmultiplikation“	
Skill	Fähigkeit		

Tabelle 9 zeigt drei mögliche Skill-Strukturen. Bei allen Ansätzen werden die Fähigkeiten auf der letzten technischen Ebene als Skills abgebildet, da diese nicht weiter unterteilt werden können.

Beim ersten Ansatz werden Studienmodule auf der Ebene von Projekten definiert. Dieser Ansatz ist daher ideal geeignet, um Themengebiete der Module (fachlich) mit den technischen Themengebieten von SkillTree zu verbinden. Auch die Unterteilung in verschiedene Lösungswege bereitet keine Schwierigkeiten.

Der zweite Ansatz beinhaltet Studiengänge auf höchster Ebene. Dies ermöglicht den Studiengangsverantwortlichen ein effizientes Management, ohne zwischen unzähligen Projekten wechseln zu müssen, wie es beim ersten Ansatz der Fall wäre. Auch ist der Fortschritt der Studierenden innerhalb ihres Studiengangs besser darstellbar. Auf der technischen Ebene der Themengebiete befinden sich bei diesem Ansatz die Studienmodule, was zur Folge hat, dass keine klare Trennung zwischen Themengebieten und Lösungswegen mehr möglich ist. Hier fehlen zusätzliche technische Ebenen für eine genaue Trennung.

Der einzige Unterschied vom zweiten zum dritten Ansatz ist die Projektebene. Auf dieser befindet sich hier ALADIN als übergreifende Lernplattform. Unter dieser werden alle Studienmodule gebündelt. Bei diesem Ansatz fällt zusätzlich zum zweiten Ansatz die Unterteilung in Studiengänge weg. Auch die Nachteile vom zweiten Ansatz bleiben bestehen. Allerdings können hier die Fortschritte aller Studierenden zwischen den verschiedenen Studiengängen und Modulen dargestellt werden.

Vor- und Nachteile der dargestellten Ansätze

Erster Ansatz (Studienmodul > Themengebiet > Fähigkeitengruppe > Fähigkeit):

Vorteile:

- Klare Zuordnung der Fähigkeiten zu Studienmodulen und Themengebieten.
- Effektives Management von fachlichen Themengebieten.
- Mögliche Differenzierung nach Lösungswegen.

Nachteile:

- Darstellung des Fortschritts nur pro Studienmodul.

Zweiter Ansatz (Studiengang > Studienmodul > Themengebiet-Lösungsweg > Fähigkeit):

Vorteile:

- Effizientes Management auf Studiengangebene.
- Klare Fortschrittsdarstellung der Studierenden im Studiengang.
- Klare Zuordnung der Fähigkeiten zu Studiengängen.

Nachteile:

- Eingeschränkte Differenzierung zwischen Themengebieten und Lösungswegen.

Dritter Ansatz (ALADIN > Studienmodul > Themengebiet-Lösungsweg > Fähigkeit):

Vorteile:

- Anzeige des Fortschritts über verschiedene Studiengänge hinweg.
- Effizientes Management durch Bündelung aller Studienmodule unter ALADIN.

Nachteile:

- Keine klare Differenzierung zwischen Studiengängen auf technischer Ebene.
- Eingeschränkte Differenzierung zwischen Themengebieten und Lösungswegen.
- Schwierigkeiten bei der Aufschlüsselung von Fortschritten auf Studiengangebene.

Fazit der dargestellten Ansätze – technische Limitationen

Der erste Ansatz zeigt die beste Übereinstimmung zwischen den technischen Strukturen von SkillTree und den fachlichen Zusammenhängen aus ALADIN, ist jedoch technisch nicht umsetzbar. Aufgrund von technischen Beschränkungen wurde der dritte Ansatz gewählt, obwohl er einige Nachteile aufweist. Die Client-Bibliotheken erlauben keine Darstellung mehrerer Projekte, daher wurde ALADIN als fachlicher und technischer Wurzelknoten gewählt, wie in den Limitationen der SkillTree Plattform im Kapitel 5.4 beschrieben. In den kommenden Kapiteln wird der dritte Ansatz umgesetzt.

7.3.3 Konzeption der Skill-Attribute

In diesem Abschnitt wird die Gestaltung der Skill-Attribute betrachtet, insbesondere die Bewertung der Punkte pro Skill und die Kriterien, die über die Höhe dieser Punkte entscheiden.

Studierende sind in der Lage, die Schwierigkeit der Aufgabe durch die anfängliche Parametrisierung zu beeinflussen. Die Betrachtung der Gesamtschwierigkeit der generierten Aufgabe wird bei diesem Ansatz nicht betrachtet. Eine Möglichkeit wäre allerdings, die Vergabe von Zusatzpunkten nach erfolgreichem Lösen der gesamten Aufgabe.

Punktebewertung pro Skill:

Die Punktezuweisung pro Skill ist ein zentraler Aspekt. Es ist wichtig, zu definieren, welche Kriterien die Höhe der Punkte bestimmen. Hierbei spielt die Schwierigkeit der Teilaufgaben eine entscheidende Rolle. Je schwieriger eine Teilaufgabe ist, desto mehr Punkte sollte der resultierende Skill bieten.

Bei der Punktebewertung ist es wichtig, die verwendeten Hilfsmethoden zu beachten.

Tabelle 10: Auswirkung der Hilfsmethoden auf die Punktzahl (Quelle: Eigene Tabelle)

Aktion	Wertung
Selbständiges Lösen der Aufgabe (keine Hilfsmethode)	kein Punktabzug
Verwendung der Kopierfunktion der Matrizen	kein Punktabzug
Lösungshilfen verwenden, bspw. „Nullen ergänzen“	Abzug der Hälfte der Punkte
Lösung anzeigen	Abzug aller Punkte

Tabelle 10 stellt die Bewertung verschiedener Hilfsmethoden im Hinblick auf die Punktevergabe dar. Zwar ist es anhand des ALADIN-Frameworks möglich, die Punkte zu berechnen, welche vor dem Verwenden von Funktionen erlangt wurden. Um die Berechnung zu vereinfachen, wird hier der Zustand betrachtet, zu dem ein definierter Pfad das erste Mal auftritt. Danach werden auf den gleichen Pfad auch keine Punkte mehr vergeben. Aufgrund einer technischen Einschränkung, die nur ganze Punkte zulässt, ist

es erforderlich, eine Lösung zur Reaktion auf Hilfsmethoden zu finden. Ein Workaround besteht darin, separate Skills inklusive Hilfen mit weniger Punkten zu bewerten.

Anzahl der Wiederholungen und Verteilung auf verschiedene Lerneinheiten:

Es wird empfohlen, dass ein Skill über mehrere Lerneinheiten hinweg wiederholt bearbeitet werden sollte, bevor er als vollständig erlernt betrachtet wird. Dieses Konzept basiert auf bewährten Lernprinzipien wie der Spaced-Repetition.

Um sicherzustellen, dass die Studierenden eine Vielfalt an Aufgaben bearbeiten und nicht nur eine bestimmte Aufgabe wiederholt lösen, wird die maximale Anzahl der Meldung eines Skills in einem festgelegten Zeitintervall festgelegt. Diese Beschränkung fördert die Abwechslung bei der Aufgabenbearbeitung und die Wiederholung von Skills über mehrere Lerneinheiten hinweg. Konkret könnte festgelegt werden, dass ein Skill ein- bis dreimal innerhalb einer Lerneinheit erfolgreich bearbeitet werden muss.

Um ein tiefgründiges und dauerhaftes Verständnis zu gewährleisten, sollten mindestens drei bis fünf Lerneinheiten durchgeführt werden. Dadurch lässt sich die Wiederholungsanzahl für den Skill bestimmen, ab welcher er als vollständig erlernt zählt.

Es ist wichtig zu betonen, dass die genaue Anzahl der Wiederholungen und der Abstand zwischen diesen, abhängig von der Art des Skills, der Komplexität der Aufgabe und den individuellen Lernbedürfnissen variieren kann.

Strukturierung und Benennung:

Die Benennung von Skill-Gruppen folgt dem Schema „Themengebiet – Lösungsweg“, um eine klare Strukturierung zu gewährleisten und den Studierenden eine gute Orientierung zu bieten.

7.3.4 Entwurf der Level als Gamification-Element in ALADIN

Die Gestaltung der Level in ALADIN ist ein wichtiger Bestandteil der Gamification-Strategie, der entscheidend für die Motivation und das Engagement der Lernenden ist. Level können durch SkillTree auf Ebene des Projektes und der Themengebiete eingesetzt werden und orientieren sich anhand der Punkte, welche im Zusammenhang mit Skills vergeben werden.

Eine zentrale Überlegung ist, wie die Level-Anpassung gestaltet werden soll. Hierbei ist die Frage zu beantworten, ob es besser ist, die Level gleichmäßig zu verteilen oder ob ein Anstieg in der Schwierigkeit angestrebt wird. Die Idee einer progressiven Schwierigkeitskurve liegt nahe, bei der die ersten Level leicht zu erreichen sind, wo aber die Herausforderung mit jedem folgenden Level signifikant steigt.

Dieser Ansatz bietet den Nutzern einen motivierenden Start. Mit jedem Levelaufstieg nimmt die Schwierigkeit dann stetig zu, was den Nutzern eine fortschreitende Herausforderung bietet und ihre Fähigkeiten und ihr Engagement fördert.

Die Idee ist, einen ausgewogenen Ansatz zu finden, der den Nutzer*innen ein Gefühl des Fortschritts vermittelt, aber sie nicht überfordert. Eine zu steile Schwierigkeitssteigerung könnte zu Frustration und Demotivation führen, während eine zu flache Steigerung das Interesse und die Herausforderung mindern könnte.

8 Implementierung der Gamification-Elemente im ALADIN-Framework

8.1 Implementierungen der Services zur Verwendung der SkillTree Plattform

SkillTree Service

Der SkillTree Service läuft als Docker Container, die einzige Voraussetzung ist ein PostgreSQL-Service. Da dieser schon beim ALADIN Backend vorhanden ist, wird SkillTree entsprechend konfiguriert und der PostgreSQL-Service kann wiederverwendet werden. Nun legt sich SkillTree eine eigene Datenbank an und erstellt alle notwendigen Tabellen und Beziehungen automatisch. Der Service kann dann direkt ohne weitere Konfiguration verwendet werden.

Authentifizierung

Damit vom Frontend Skills gemeldet werden können, müssen die jeweiligen Clients authentifiziert werden. Aktuell verwendet ALADIN selbst noch keine Authentifizierung, weshalb ein statischer Benutzer verwendet wird. Der Authentifizierungsservice benötigt einen Admin-Zugang für den SkillTree Service, welcher per Konfiguration eingelesen wird.

Implementiert ist dieser Service anhand eines Beispiels in Java-Spring-Boot. Er stellt eine Schnittstelle zum Authentifizieren bereit („/users/{user}/{projectId}/token“). Diese Schnittstelle verwendet den Client Credentials Flow im OAuth 2.0-Protokoll, welcher keine weiteren Benutzeranmeldeinformationen (wie Passwort) erfordert. Stattdessen verwendet er die Client-ID und das Client-Secret. Die Client-ID entspricht der Projekt-ID, aus der URL. Das Client-Secret muss der Service unter Verwendung des Admin-Zugangs vom SkillTree Service anfragen. Mit Client-ID und Client-Secret kann nun ein Access-Token an den Client gereicht und dort als Cookie verwendet werden.

Versuchen Clients auf Daten des SkillTree Services zuzugreifen, müssen sie sich erst authentifizieren. Der Authentifizierungsservice liefert einen Session-Cookie für das jeweilige Projekt, mit welchem die Clients dann auf die korrekten Daten zugreifen können.

Datenimport

Der Datenimport dient dazu, den SkillTree Service automatisch mit der notwendigen Skill-Struktur zu befüllen. Er läuft einmalig und stoppt danach wieder. Implementiert ist er in Java-Spring-Boot.

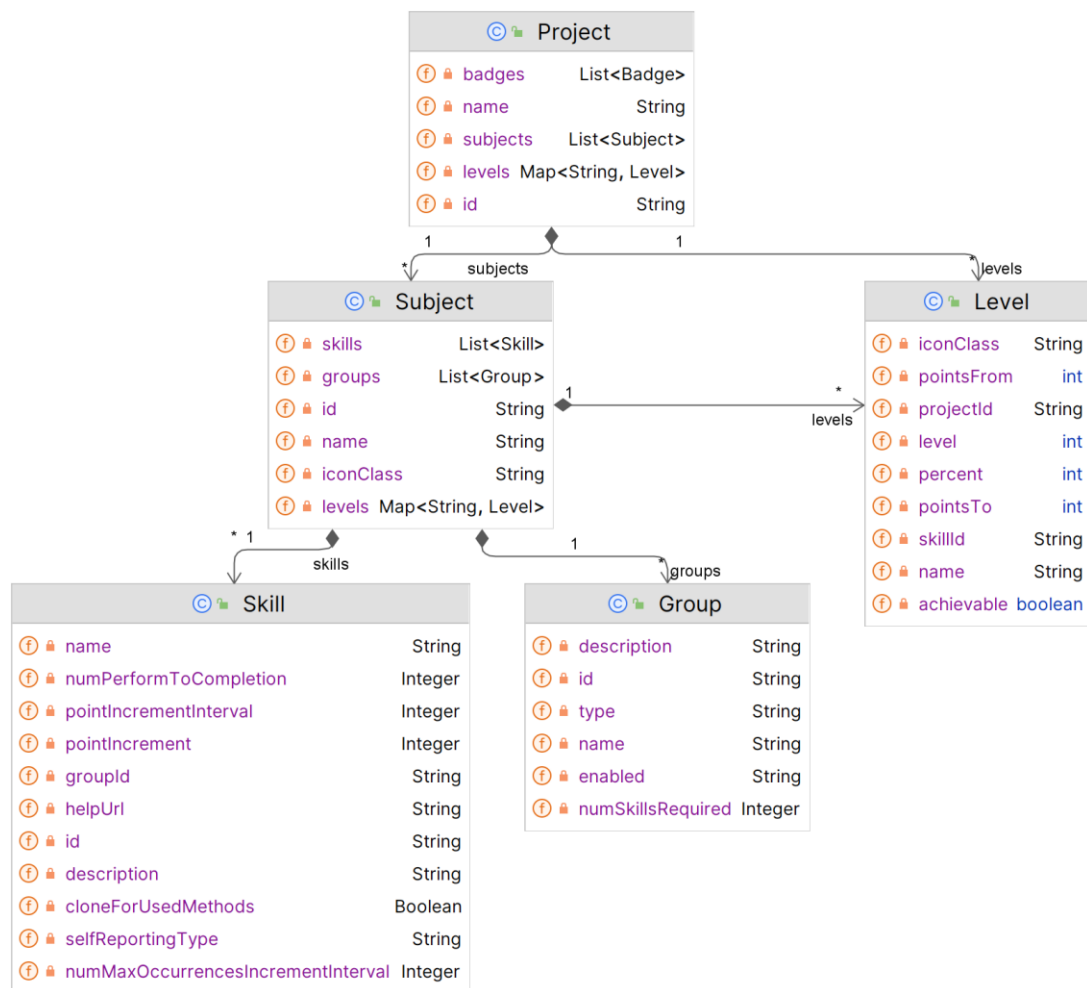


Abbildung 10: Java Struktur des SkillTree Imports (Quelle: Eigene Darstellung)

Abbildung 10 zeigt die implementierten Klassen, um die notwendige Struktur für den Import abzudecken. Dargestellt werden auch alle Attribute der Klassen, wobei einige optional sind. Im Folgenden wird nur auf die wichtigsten Klassen und Attribute genauer eingegangen.

Das Projekt (Project) steht an oberster Stelle, dieses beinhaltet sowohl alle Themengebiete (Subjects) in Form einer Liste, als auch die entsprechend definierten

Level als Schlüssel-Wert-Paare. Außerdem werden mögliche Abzeichen (Badges) in Form einer Liste abgelegt. Im Weiteren werden die Badges nicht genauer aufgeführt und erläutert, da sich die Implementierung auf die Skills und Level beschränkt.

Levels werden als Schlüssel-Wert-Paare abgelegt, wobei der Schlüssel die Nummer des Levels ist. Zusätzlich müssen noch der Name des Levels und der prozentuale Anteil an der Gesamtpunktzahl festgelegt werden. Die Gesamtpunktzahl bezieht sich auf die Punktzahl des Projektes oder der Themengebiete, für die Level definiert werden. SkillTree passt die Level dann dynamisch an die maximale Punktzahl an, sobald die Punktzahl der Skills nachträglich angepasst wird.

Themengebiete stellen Studienmodule dar und dienen dem Gruppieren der Skills, daher benötigen diese nur einen Namen und beinhalten selbst die Skills und Gruppen als Listen.

Gruppen werden verwendet, um Skills von Themengebieten und Lösungswegen zusätzlich zu gruppieren. Sie selbst benötigen nur einen Namen und die Anzahl der mindestens abgeschlossenen Skills, welche zum erfolgreichen Abschließen der Gruppe benötigt werden.

Skills benötigen einen Namen, die Anzahl der Punkte pro Meldung, die Anzahl der Wiederholungen und Informationen zur Beschränkung der Meldungen pro Lerneinheit. Zusätzlich kann der Identifikator einer Gruppe definiert werden, zu der dieser Skill zugeordnet werden soll. Auf Datenbankebene befindet sich zwischen Gruppe und dem Skill eine 1 zu n Beziehung. Da bei der Definition des JSON allerdings keine Integritätsprüfung durchgeführt wird, wird in Abbildung 10 keine Beziehung dargestellt. Um den Definitionsaufwand gering zu halten, gibt es die Möglichkeit, automatisch beim Import einen Klon mit der Hälfte der Punkte anzulegen. Dieser Klon kann verwendet werden, um Aufgaben zu bewerten, bei denen Hilfsmethoden verwendet wurden.

Anhand dieser Klassen-Struktur kann nun eine JSON-Datei erstellt werden, in der die genaue Struktur für SkillTree definiert wird. Durch Anfragen an die REST-Schnittstellen des SkillTree Services werden in diesem die Elemente auf Datenbankebene erstellt.

8.2 Implementierungen im Backend des ALADIN-Frameworks

Im Backend des ALADIN-Frameworks musste die Aufgabe.json angepasst werden. Bisher war es der Fall, dass Hilfsmethoden für die jeweiligen Oberflächenelemente nur mit einer Beschreibung definiert wurden. Nun muss zusätzlich die Auswirkung auf die Punkte definiert werden. Wird beispielsweise die Hilfsmethode „Nullen ausfüllen“ verwendet, so werden automatisch Felder befüllt, die Studierenden bekommen also nicht die volle Punktzahl.

Um das zu erreichen, muss die Struktur der Aufgabe.json erweitert werden. Statt eines einfachen Beschreibungsfeldes wird ein Objekt eingeführt, in welchem sowohl die Beschreibung als auch die Auswirkung auf die Punkte definiert werden.

8.3 Implementierung im Frontend des ALADIN-Frameworks

Anpassung an die neue Methodenstruktur

Da Methoden durch die Anpassung im Backend in einem anderen Format definiert werden, muss auch das Frontend angepasst werden. Alle Basiselemente können zusätzliche Methoden beinhalten, somit wurden Matrix-Elemente, Gantt-Diagramm-Elemente und Graphen-Elemente abgeändert. Zusätzlich wurde eine zentrale Methode implementiert, welche die Methodendefinitionen aus der Aufgabe.json ausliest und in dem gewünschten Format für alle Elemente bereitstellt.

Definition der Schnittstelle

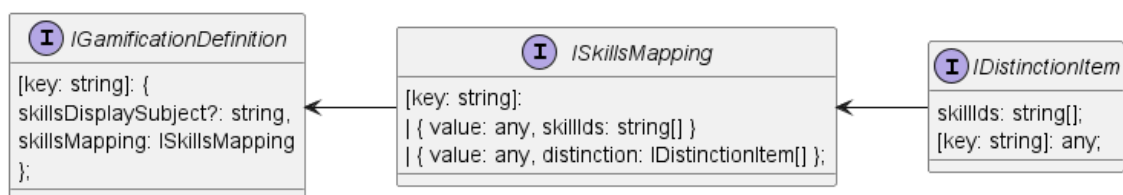


Abbildung 11: Programmierte Struktur der Gamificationdefinition im Frontend (Quelle: Eigene Darstellung)

Die Zuordnung der Pfade aus der Aufzeichnung zu den Skills im SkillTree Service erfolgt anhand der Struktur, welche in Abbildung 11 dargestellt wird. In TypeScript werden drei Interfaces angelegt, welche die gewünschte Datenstruktur beschreiben.

Jede Aufgabe in ALADIN erhält ein zugeordnetes SkillsDisplaySubject. Dieser Wert

wird später für die automatische Navigation in der SkillTree-Frontend-Ansicht verwendet. Dadurch werden den Studierenden immer die Fähigkeiten der aktuellen Aufgabe angezeigt. Ein selbständiges Navigieren in der Ansicht entfällt.

Die Gamificationdefinition enthält ein SkillsMapping, welches die Zuordnung von Pfaden zu den zugehörigen Skills regelt. Diese Zuordnung bildet die Grundlage für die Skill-Verknüpfungen im SkillTree Service.

Innerhalb des SkillsMapping gibt es zwei Varianten:

1. Pfad, Wert und SkillIds:

Hier wird einem bestimmten Pfad ein Wert zugeordnet und es können ein oder mehrere SkillIds mit diesem Pfad verbunden werden. Dies ermöglicht eine direkte Zuweisung von Lernfortschritten zu bestimmten Skills.

2. Regex-Pfad, Wert und DistinctionItem:

Eine alternative Struktur nutzt Regex-Pfade, um eine flexiblere Zuordnung zu ermöglichen. Neben dem Wert werden hier weitere Unterscheidungsmerkmale in einem separaten Objekt definiert. Auch für diese Variante können SkillIds zugeordnet werden, um eine differenzierte Skill-Verbindung herzustellen.

Abbildung A20 im Anhang zeigt eine beispielhafte Definition und Darstellung der Zuordnung.

Implementierung der Skill-Meldefunktionalität

Die Implementierung der Skill-Meldefunktionalität umfasst mehrere wichtige Funktionen, die für die erfolgreiche Zuordnung von Fähigkeiten zu Pfaden und Oberflächenelementen erforderlich sind:

1. GetComponent:

Diese Funktion liefert den aktuellen Status des Oberflächenelements, anhand von einem Pfad. Sie dient als Grundlage für die Zustandsbestimmung des Elements.

2. matchesDistinction:

Diese Funktion prüft die zusätzlichen Unterscheidungsmerkmale, des DistinctionItem, anhand des Oberflächenelement-Status, welcher durch die Funktion GetComponent abgerufen werden kann. Sie gibt „wahr“ zurück, wenn die Überprüfung erfolgreich ist, andernfalls „falsch“.

3. getSkillIds:

Diese Funktion bestimmt die definierten SkillIds für einen gegebenen Pfad. Entweder ist ein Pfad eindeutig mit SkillIds verknüpft oder es wird die Funktion matchesDistinction verwendet, falls der Pfad als regulärer Ausdruck angegeben ist.

4. getMethods:

Diese Funktion liefert alle bisher verwendeten Methoden für das aktuelle Element. Hierfür wird der Status des Oberflächenelements benötigt. Die Kenntnis der verwendeten Methoden ist essenziell für die Bewertung der Fähigkeit.

5. getHighestImpactOfMethods:

Diese Funktion ermittelt die höchste Auswirkung der verwendeten Methoden. Diese Auswirkung ist ein entscheidender Faktor bei der Punktevergabe.

6. submitSkills:

Diese Funktion verwendet direkt oder indirekt alle zuvor implementierten Funktionen und dient dazu, einen Skill an den SkillTree Service zu melden, sofern eine SkillId für den Pfad vorhanden ist. Sie ist der Schlüsselmechanismus, um die erworbenen Fähigkeiten zu übermitteln und im SkillTree zu registrieren.

Die Funktion "submitSkills" wird im Hintergrund asynchron ausgeführt, sobald der Benutzer Aktionen durchführt. Zusätzlich wird festgehalten, welche Pfade bereits bearbeitet wurden, damit für den gleichen Pfad nur einmal Punkte vergeben werden. Vor dem Absenden wird geprüft, wie stark die verwendeten Hilfsmethoden die Punktzahl beeinflussen. Wenn mehr als die Hälfte der Punkte abgezogen wird, wurde eine große Lösungshilfe verwendet und es werden keine Skills übermittelt. Wird maximal die Hälfte

der Punkte abgezogen, werden die geklonten Skills mit weniger Punkten gemeldet. Falls keine Auswirkung der Hilfsmethoden auf die Punkte besteht, werden die normalen Skills übermittelt.

Integration der SkillTree-Ansicht in das Frontend

Die von SkillTree bereitgestellte Skill-Ansicht soll im ALADIN-Frontend integriert werden, damit die Benutzer ihren Fortschritt sehen können. Dafür wird eine Schaltfläche in das Frontend integriert, nach dessen Betätigung die Ansicht angezeigt wird. Zu beachten ist, dass diese Ansicht nicht als separate Seite angezeigt werden soll, sondern als Popup über der bestehenden. Dadurch können Studierende ihren Fortschritt während der Aufgabenbearbeitung verfolgen.

Innerhalb der Schaltfläche wird das aktuelle Level des Benutzers über alle Module hinweg angezeigt. Dafür wurde eine vorgefertigte Komponente von SkillTree verwendet, leider lässt diese keine individuelle Levelanzeige pro Studienmodul zu.

Zum Anzeigen der Skill-Ansicht wird die Modal-Komponente von Bootstrap verwendet, diese „schwebt“ dann über der aktuellen ALADIN-Ansicht. Da sich die Skill-Ansicht von SkillTree automatisch an die vorhandene Breite anpasst, kann sie mühelos integriert werden. Zusätzlich lässt sich die Skill-Ansicht weiter konfigurieren, so wird in diesem Fall die interne „Zurück-Schaltfläche“ aktiviert, welche den Studierenden eine einfachere Navigation ermöglicht.

Unterschiedliche Aufgaben in ALADIN sind per Gamificationdefinition anderen Studienmodulen zugeordnet, für diese gibt es in der Skill-Ansicht jeweils andere Ansichten. Damit Studierende nicht händisch beim Lösen von Aufgaben in der Skill-Ansicht navigieren müssen, wird programmatisch anhand des SkillsDisplaySubject das Studienmodul zur aktuellen Aufgabe angezeigt.

Implementierung des Live-Feedbacks

Werden an SkillTree Skills gemeldet, kann der Status dieser Meldung abgefragt werden, dadurch lässt sich ein Live-Feedback implementieren. Dieses Feedback wird in Form einer Bootstrap Toast-Komponente angezeigt, die dem Benutzer die gemeldete Fähigkeit sowie die Anzahl der verdienten Punkte präsentiert. Wurde ein Skill innerhalb des

aktuellen Intervalls zu häufig gemeldet, wird ein entsprechender Hinweis dargestellt. Die Implementierung des Live-Feedbacks trägt dazu bei, dass die Benutzer ihre Fortschritte klar erkennen können, was die Motivation und damit die Effizienz des Lernprozesses steigert.

8.4 Herausforderungen bei der Implementierung der Gamification-Elemente

Die Implementierung der Gamification-Elemente im ALADIN-Framework stellte einige Herausforderungen dar, die erfolgreich bewältigt wurden, aber dennoch den Entwicklungsprozess beeinflussten.

1. Einarbeitung in das SkillTree Framework: Um die Gamification-Elemente effektiv zu integrieren, war eine gründliche Einarbeitung in das SkillTree Framework notwendig. Dies ermöglichte die korrekte Verwendung der bereitgestellten Komponenten für eine reibungslose Interaktion mit dem SkillTree Service.
2. Einarbeitung in das ALADIN Framework: Das ALADIN-Framework ist komplex und erforderte eine eingehende Einarbeitung, um die Zusammenhänge und Logik zu verstehen. Besonders die konfigurative Programmierung in ALADIN stellte eine Herausforderung dar.
3. Kontinuierliche Veränderungen an ALADIN: Die fortlaufenden Änderungen und Weiterentwicklungen im ALADIN-Framework führten zu Veränderungen in der Struktur der Aufzeichnungen, die als zentrale Schnittstelle für die Gamification-Elemente dienten. Diese Veränderungen erforderten eine kontinuierliche Anpassung und Aktualisierung der Implementierung der Gamification-Elemente.

8.5 Endzustand der Infrastruktur zur Gamification der ALADIN Lernplattform unter Verwendung von SkillTree

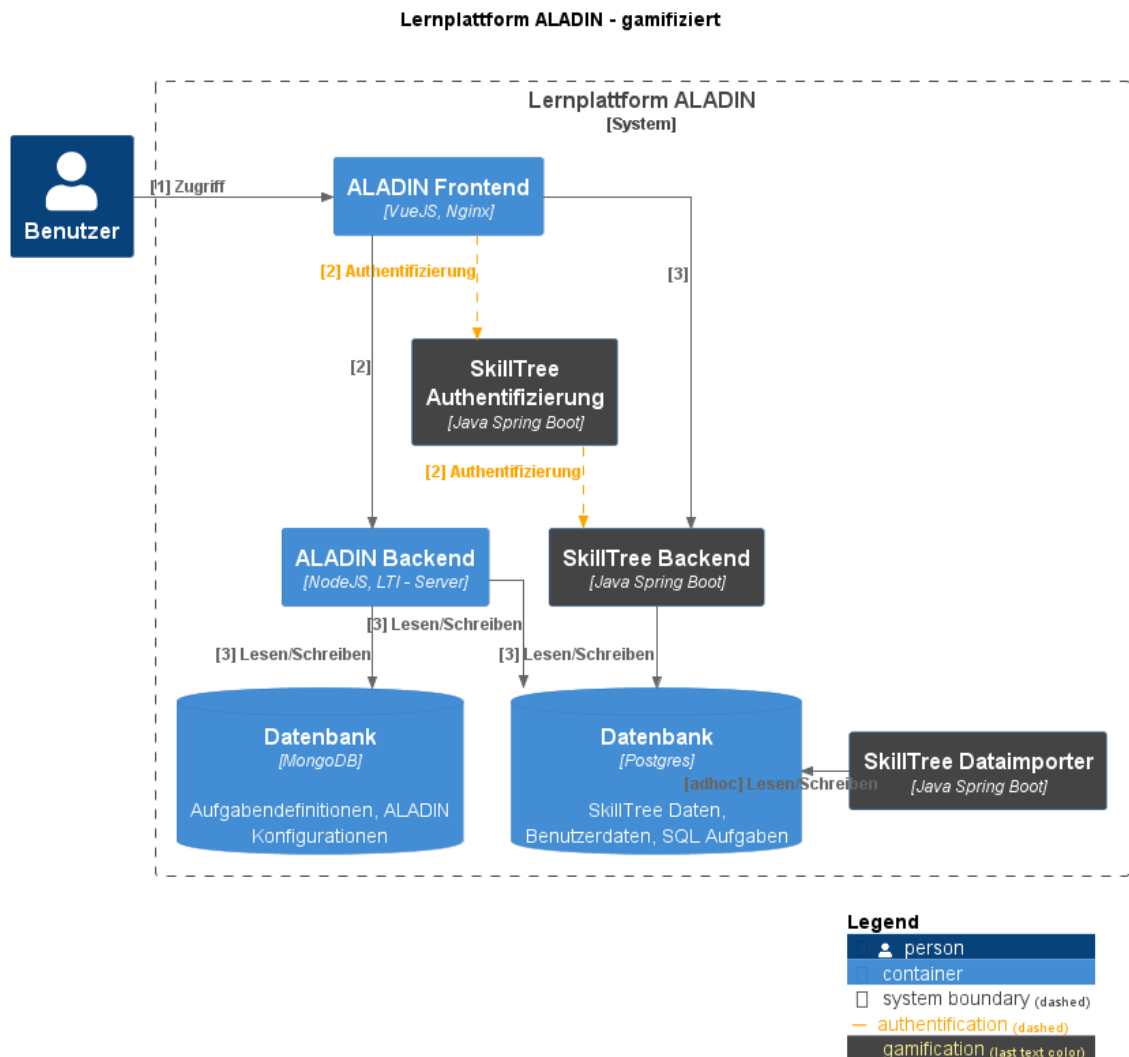


Abbildung 12: Endzustand der gamifizierten Lernplattform ALADIN (Quelle: Eigene Darstellung)

In Abbildung 12 wird der Endzustand der Lernplattform dargestellt. Die dunklen Elemente stellen zusätzliche Services für die Gamification dar. Zusätzlich wurde das ALADIN-Frontend um Funktionen und Komponenten erweitert. Der PostgreSQL-Service wurde um eine weitere Datenbank für SkillTree erweitert. Diese Infrastruktur bietet eine gute Skalierbarkeit und Konfigurierbarkeit, was zukünftige Anpassungen erleichtert.

9 Test der Gamification-Elemente im ALADIN-Framework

Aufgrund zeitlicher Einschränkungen können keine Nutzertests zur Wirkungsbelegung der Gamification-Elemente in dieser Bachelorarbeit durchgeführt werden. Deshalb wird anhand von manuellen Tests die Funktionalität der implementierten Services, Funktionen, Elemente und der SkillTree Plattform überprüft.

9.1 Test des Skill-Systems/Punktesystems als Gamification-Element im ALADIN-Framework

9.1.1 Testentwurf

Für den Test des Punktesystems werden repräsentative Aufgaben aus verschiedenen Fachgebieten ausgewählt. Für diese Aufgaben werden Skills für den Import in SkillTree definiert und eine Gamificationdefinition mit spezifischen Pfaden angelegt. Anschließend werden alle Services gestartet und eine Testperson bearbeitet die Aufgaben. Währenddessen wird überprüft, ob die Punktevergabe korrekt funktioniert. Konkret wird getestet:

- Werden Punkte für korrekte Lösungsschritte vergeben?
- Stimmt die Höhe der vergebenen Punkte mit der Konfiguration überein?
- Werden bei Verwendung von Hilfsfunktionen Punkte abgezogen?
- Funktioniert die Übertragung der Punkte an den SkillTree Service korrekt?

9.1.2 Testdurchführung

Die Konfiguration der Skills inklusive der Höhe der Punkte findet im SkillTree Datenimporter statt, konkret ist diese in Abbildung A21 und Abbildung A22 des Anhangs dargestellt. Die Gamificationdefinition, welche als Schnittstelle dient, wird im ALADIN-Frontend umgesetzt und ist in Abbildung A25 und Abbildung A26 im Anhang zu sehen.

Die Tests werden anhand der Stücklistenauflösung und der Netzplantechnik durchgeführt. Pro Aufgabe werden mehrere Durchläufe mit verschiedenen Lösungsvarianten (korrekt, falsch, mit/ohne Hilfen) gemacht. Außerdem wird das Zeitintervall zum Verteilen auf mehrere Lerneinheiten getestet. Teile der Durchläufe werden durch Abbildung A27 bis Abbildung A35 im Anhang dargestellt. Die

Punktevergabe wird überprüft.

9.1.3 Testauswertung

Die Tests zeigen, dass die Punktevergabe bei korrekter Lösung konsistent mit der Konfiguration funktioniert. Bei Fehler oder Hilfen werden die Punkte wie definiert abgezogen. Die Übermittlung an SkillTree ist erfolgreich.

9.2 Test der Level als Gamification-Element im ALADIN-Framework

9.2.1 Testentwurf

Für den Test der Level wird die korrekte Berechnung und Anzeige der Level in Abhängigkeit von den gesammelten Punkten überprüft. Dafür werden unterschiedliche Level auf ALADIN und auf Studienmodulebene definiert. Anschließend werden alle Services gestartet und eine Testperson bearbeitet Aufgaben, um Punkte zu sammeln. Folgendes wird getestet:

- Werden die Level korrekt berechnet, basierend auf den gesammelten Punkten?
- Erfolgt die Anzeige des Levels im ALADIN-Frontend korrekt?

9.2.2 Testdurchführung

Die Konfiguration der Level erfolgt im SkillTree Datenimporter, die für diesen Test definierten Level werden in den Abbildungen Abbildung A23 und Abbildung A24 im Anhang dargestellt.

In mehreren Testdurchläufen werden gezielt Punkte gesammelt, um die Berechnung und Anzeige der Level zu testen, dargestellt in Abbildung A27 bis Abbildung A35 im Anhang. Der aktuelle Levelstand wird überprüft, die Anzeige der verschiedenen Level für ALADIN und die Studienmodule wird in Abbildung A36 bis Abbildung A38 im Anhang abgebildet.

9.2.3 Testauswertung

Die Tests zeigen eine korrekte Berechnung und Darstellung der Level in Abhängigkeit von den Punkten.

10 Fazit und Ausblick

10.1 Zusammenfassung der Ergebnisse

In dieser Arbeit wurden die Gamification-Elemente Punktesystem und Level exemplarisch in der Lernplattform ALADIN entwickelt und implementiert. Die Elemente wurden auf Basis der Anforderungen der Nutzergruppen konzipiert und unter Verwendung der SkillTree Plattform technisch realisiert. Dabei hat sich gezeigt, dass durch das Skill-System/Punktesystem viele weitere Gamification-Elemente, wie Ranglisten, abgeleitet werden können. Die Implementierung erfolgte durch Erweiterung der ALADIN Software-Architektur. Dabei hat sich ALADIN als geeignete Plattform für die Integration von Gamification erwiesen.

10.2 Beantwortung der Forschungsfragen

Bezugnehmend auf die Forschungsfragen aus Kapitel 1.1 lassen sich diese wie folgt beantworten:

Zentrale Forschungsfrage:

„In welcher Weise können Gamification Elemente in der Lernplattform ALADIN entwickelt und implementiert werden, um die Lernmotivation und das Lernverhalten der Studierenden nachhaltig zu verbessern?“

Die exemplarische Entwicklung in dieser Arbeit zeigt, dass sich die Gamification-Elemente Punktesystem und Level durch die Integration mit der SkillTree Plattform effektiv in ALADIN integrieren lassen. Durch die Verknüpfung der Aufgabenpfade mit Skills konnte eine flexible Lösung geschaffen werden, die sich gut an die fachlichen Inhalte und Strukturen von ALADIN anpasst.

Unterfragen:

„Welche Erfolgsfaktoren gibt es bei der Konzeption von Gamification-Elementen?“

Als zentrale Erfolgsfaktoren wurden in Kapitel 2.6 unter anderem die passende Kombination der Elemente für die Zielgruppe, der konkrete Nutzen für die Anwender, die Vermeidung von Überforderung sowie ein angemessenes Feedbacksystem

identifiziert.

„Welche Ansätze eignen sich für die Implementierung der Gamification-Elemente in ALADIN?“

In Kapitel 7.2 und 7.3 wurden verschiedene technische Ansätze zur Integration analysiert. Als geeignet wurde die Verknüpfung der Aufgabenpfade mit Skills der SkillTree Plattform befunden, da dies eine flexible und erweiterbare Lösung ermöglicht.

„Welche Herausforderungen gibt es bei der Implementierung der Gamification Elemente?“

Kapitel 8.4 zeigt Herausforderungen, wie die Einarbeitung in die komplexen Frameworks von ALADIN und SkillTree. Insgesamt konnten die Gamification-Elemente aber erfolgreich integriert werden.

10.3 Implikation für die Praxis

Die exemplarische Entwicklung liefert einen Leitfaden und Referenzimplementierung für die Gamification von Lernplattformen. Sie zeigt konkrete technische Lösungsansätze und Herangehensweisen auf. Die Ergebnisse können von anderen Lernplattformen übernommen werden und als Basis für Erweiterungen dienen.

10.4 Kritische Reflexion und Limitation der Arbeit

Eine Limitation besteht darin, dass die Wirkung der implementierten Elemente auf die Lernmotivation noch nicht abschließend evaluiert wurde. Hier wären Nutzerstudien sinnvoll. Die Bindung an SkillTree könnte bei stark veränderten Anforderungen Nachteile bringen. Insgesamt liefert die Arbeit aber einen guten Ausgangspunkt.

10.5 Ausblick und zukünftige Forschungen

Ein entscheidender Schritt für die effektive Nutzung der Gamification-Elemente ist die Implementierung einer Login-Funktion, um die Punkte den jeweiligen Benutzern zuzuordnen. Eine weitere Herausforderung besteht darin, die individuelle Schwierigkeit der parametrisierten Aufgaben zu bestimmen. Dadurch könnte die Höhe der Punkte angepasst werden. Eine flexible Vergabe von Punkten beim Melden der Skills wäre

vorteilhaft. Die Konzeption weiterer Gamification-Elemente wie Badges oder Feedback-Systeme wird empfohlen. Außerdem wäre es sinnvoll, die Funktionalität der Lernpfade in SkillTree zu testen. Abschließend müssen die tatsächlichen Motivationseffekte durch Studien analysiert werden.

11 Literaturverzeichnis

- Alsawaier, R., 2017. The Effect of Gamification on Motivation and Engagement. *International Journal of Information and Learning Technology*, November.
- Balzert, H., 1998. *Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering*. 3 Hrsg. Heidelberg: Spektrum.
- Bartle, R., 1996. Hearts, clubs, diamonds, spades: Players who suit MUDs. *Journal of MUD Research*.
- Bitkom, 2022. Anteil der Computer- und Videospieler in verschiedenen Altersgruppen in Deutschland im Jahr 2022. Statista. Statista GmbH. [Online] Verfügbar unter: <https://de.statista.com/statistik/daten/studie/315924/umfrage/anteil-der-computerspieler-in-deutschland-nach-alter/> [Zugriff am 30 September 2023].
- Blum, W., 2007. *Modellierungsaufgaben im Mathematikunterricht – Herausforderung für Lehrerinnen und Lehrer*. Kassel: s.n.
- Borukhovich-Weis, S., Gryl, I., Łączkowska, E. & Bulizek, B., 2021. *Mobiles Lernen, Selbststeuerung und Gamification. Ergebnisse einer qualitativen Begleitstudie zu Chancen und Grenzen einer Inverted-Classroom-Veranstaltung in der Lehrer*innenprofessionalisierung*. s.l.:Hochschulforum Digitalisierung.
- Christ, P., Laue, R. & Munkelt, T., 2022. ALADIN – Generator für Aufgaben und Lösung(shilf)en aus der Informatik und angrenzenden Disziplinen. *Gesellschaft für Informatik*.
- Korn, O., Schulz, A. S. & Hagley, B. J., 2022. *Gamification: Grundlagen, Methoden und Anwendungsbeispiele*. s.l.:Springer.
- Liarokapis, F., Anderson, E. F. & Oikonomou, A., 2010. *Serious Games for use in a Higher Education Environment*. [Online] Verfügbar unter: https://www.researchgate.net/publication/230778862_Serious_Games_for_use_i

n a Higher Education Environment

[Zugriff am 24 Mai 2023].

Munkelt, T. & Christ, P., 2022. *ALADIN II -GENERATOR FÜR AUFGABEN DISZIPLINEN II. 20. Workshop on e-Learning..* Zittau/Görlitz, s.n.

Pawelka, F., Wollmann, T., Stöber, J. & Lam, T. V., 2017. *Erfolgreiches Lernen durch gamifiziertes E-Learning.* [Online]

Verfügbar unter: <https://dl.gi.de/handle/20.500.12116/2874>

[Zugriff am 17 Juli 2023].

Ryan, R. M. & Deci, E. L., 2000. Self-Determination Theory and the Facilitation of Intrinsic Motivation, Social Development, and Well-Being. *American Psychologist*, Januar.

Sailer, M., 2016. *Die Wirkung von Gamification auf Motivation und Leistung.* München: Springer.

Stieglitz, S., 2015. Gamification – Vorgehen und Anwendung. *HMD Praxis der Wirtschaftsinformatik*, 1 Oktober, p. 816–825.

Stieglitz, S., Lattemann, C. & Christoph, G., 2010. *Learning arrangements in virtual worlds. Tagungsband 43. Hawaii International Conference on System Science (HICSS).* Hawaii, s.n.

Strahringer, S. & Leyh, C., 2017. *Gamification und Serious Games.* [Online]

Verfügbar unter: https://link.springer.com/chapter/10.1007/978-3-658-16742-4_1

[Zugriff am 10 Mai 2023].

Tolks, D. & Sailer, M., 2021. *Gamification als didaktisches Mittel.* s.l.:Hochschulforum Digitalisierung.

Wilmers, A., Achenbach, M. & Keller, C., 2021. *Bildung im digitalen Wandel Organisationsentwicklung in Bildungseinrichtungen.* [Online]

Verfügbar unter: <https://elibrary.utb.de/doi/book/10.31244/9783830994558>

[Zugriff am 17 Juni 2023].

12 Eigenständigkeitserklärung

Hiermit erkläre ich, dass ich die vorgelegte Arbeit mit dem Titel: „Exemplarische Entwicklung von Gamification-Elementen in der Lernplattform ALADIN“ selbständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie alle wörtlich oder sinngemäß übernommenen Stellen in der Arbeit als solche und durch Angabe der Quelle gekennzeichnet habe. Dies gilt auch für Zeichnungen, Skizzen, bildliche Darstellungen sowie für Quellen aus dem Internet.

Mir ist bewusst, dass die Hochschule für Technik und Wirtschaft Dresden Prüfungsarbeiten stichprobenartig mittels der Verwendung von Software zur Erkennung von Plagiaten überprüft.

Ort, Datum

Unterschrift Student/Studentin

13 Anhang

13.1 Anforderungen an die Lernplattform ALADIN

Die nachfolgenden Anforderungen werden in funktionale und nicht-funktionale Anforderungen unterteilt. Hierbei beschreiben funktionale Anforderungen jene, die direkt mit dem Zweck des Systems zusammenhängen. Nicht-funktionale Anforderungen beschreiben Restriktionen, die sich nicht auf einzelne Funktionen beziehen, sondern auf das gesamte System gelten, hierunter zählen vor allem auch Qualitätseigenschaften. (Balzert, 1998)

Funktionale Anforderungen:

- Generierung von Übungs- und Modellierungsaufgaben aus der Informatik und angrenzenden Disziplinen
- Parametrisierung der Aufgaben (Schwierigkeitsgrad)
- Unterstützung durch Lösungshinweise und Teillösungen
- Aufzeichnung und Wiedergabe der Lösungsversuche

(Christ, et al., 2022)

Nicht-funktionale Anforderungen:

- Übersichtlichkeit
- Bedienbarkeit
- Multilingualität
- Reaktivität

13.2 Designprinzipien zum Einsatz von Gamification in der Hochschulbildung nach (Tolks & Sailer, 2021)

Tabelle A11: Designprinzipien zum Einsatz von Gamification in der Hochschulbildung (Quelle: Eigen Tabelle)

Designprinzip	Beschreibung
Kombination von Game-Design-Elementen	Dieses Prinzip beinhaltet die Nutzung mehrerer Elemente, um unterschiedliche Motivationsfaktoren und psychologische Grundbedürfnisse anzusprechen.
Aktivierendes Lernen	Aktives Einbeziehen der Studierenden durch Maßnahmen wie Quizprogramme oder Wettbewerb dieser „kann die Aktivierung der Studierenden über die komplette Lerneinheit“ (Tolks & Sailer, 2021) zur Folge haben.
Theoretische Fundierung	Hierbei erfolgt der Einsatz von Gamification basierend auf theoretischen Grundlagen wie der Selbstbestimmungstheorie (Ryan & Deci, 2000) und der Theorie des Gamified Learning.
Abwägung von Wettbewerb	Dieses Prinzip betont die Notwendigkeit, das Risiko von Stress und Ungleichheit im Zusammenhang mit Wettbewerben zu berücksichtigen.
Kooperative Aufgaben, Teams	Die Nutzung kooperativer Aufgaben und Teams kann die Motivation steigern, kollaboratives Lernen fördern und den Leistungsdruck reduzieren.
Storytelling	Narrative Elemente werden verwendet, um den Transfer von Wissen und die Motivation zu fördern. Dies macht den Lernkontext interessanter.
Vernetzung mit Inverted	Dieses Prinzip bezieht sich auf die Motivation während

Classroom	der Onlineselbstlernphase und die interaktivere Gestaltung der Präsenzphase durch beispielsweise Quizformate.
Nachhaltigkeit	Der Einsatz von Gamification erfordert zusätzlichen Aufwand bei der Entwicklung und Konfiguration, daher ist er vorrangig für fortlaufende Lehrveranstaltungen und Pflichtveranstaltungen geeignet.
Differenzierung	Es ist wichtig zu prüfen, welche spielbasierten Methoden (Game-based Learning, Serious Games, Lern- und Planspiele) für welchen Zweck eingesetzt werden sollen.
Neue Gamification-Elemente	Über die traditionellen Elemente wie Punkte, Badges und Leaderboards hinaus sollten neue Elemente entwickelt und verwendet werden.
Anpassung an Zielgruppe	Die Auswahl und Nutzung der Gamification-Elemente sollte auf die jeweiligen Spielertypen und die Zielgruppe zugeschnitten sein.
Keine pauschale Wirksamkeit	Es ist zu beachten, dass die Wirksamkeit von Gamification nicht immer den Erwartungen entspricht und stark von der konkreten Umsetzung abhängig ist. Die Herausforderung und die Möglichkeit zur Bewältigung von Aufgaben spielen eine entscheidende Rolle.

13.3 Einsatz von Gamification in unterschiedlichen Kontexten

Tabelle A12: Übersicht Kontexte empirischer Studien (Sailer, 2016)

Kontext	Anzahl	Quellen
Arbeit	9	Farzan, DiMicco, Millen, Dugan, et al. (2008), Farzan, DiMicco, Millen, Brownholtz, et al. (2008), Jung, Schneider und Valacich (2010), Flatla, Gutwin, Nacke, Bateman und Mandryk (2011), Passos et al. (2011), Bagley (2012), Depura und Garg (2012), Fernandes et al. (2012), Thom, Millen und DiMicco (2012), Arai et al. (2014)
Bildung	17	Halan, Rossen, Cendan und Lok (2010), Fitz-Walter, Tjondronegoro und Wyeth (2011, 2012), Gåsland (2011), Smith und Baker (2011), Dong et al. (2012), Foster et al. (2012), Li, Grossman und Fitzmaurice (2012), McDaniel, Lindgren und Friskics (2012), Cheong et al. (2013), Denny (2013), Domínguez et al. (2013), Hakulinen et al. (2013), Snyder und Hartig (2013), Banfield und Wilkerson (2014), Landers und Landers (2014), Shi, Cristea, Hadzidedic und Dervishalidovic (2014), Goehle (2013)
Crowdsourcing	7	Liu et al. (2011), Witt, Scheiner und Robra-Bissantz (2011), Eickhoff, Harris, de Vries und Srinivasan (2012), Mason, Michalakidis und Krause (2012), Massung, Coyle, Cater, Jay und Preist (2013), Mekler et al. (2013a), Mekler et al. (2013b)
Datenerhebungen und Umfragen	3	Musthag, Raij, Ganesan, Kumar und Shiffman (2011), Downes-Le Guin, Baker, Mechling und Ruyle (2012), Rapp, Marcengo, Console und Simeoni (2012)
Gesundheit	7	Cafazzo, Casselman, Hamming, Katzman und Palmert (2012), Hamari und Koivisto (2013), Hori, Tokuda, Miura, Hiyama und Hirose (2013), K. J. Rose, König und Wiesbauer (2013), Stinson et al. (2013), Jones et al. (2014), Koivisto und Hamari (2014)
Marketing	2	Hamari (2013), Hamari (2015)
Online Communities und soziale Netzwerke	7	Montola et al. (2009), Farzan und Brusilovsky (2011), Cramer, Rost und Holmquist (2011), Bista, Nepal und Paris (2012), Bista, Nepal, Colineau und Paris (2012), Frith (2012), Grant und Betts (2013), Anderson et al. (2013)
Umweltschutz	4	Gustafsson et al. (2009), Liu et al. (2011), Gnauk, Dannecker und Hahmann (2012), Berengueres, Alsuwairi, Zaki und Ng (2013)

13.4 Ablauf der Stücklistenauflösung in der ALADIN-Lernplattform

1.

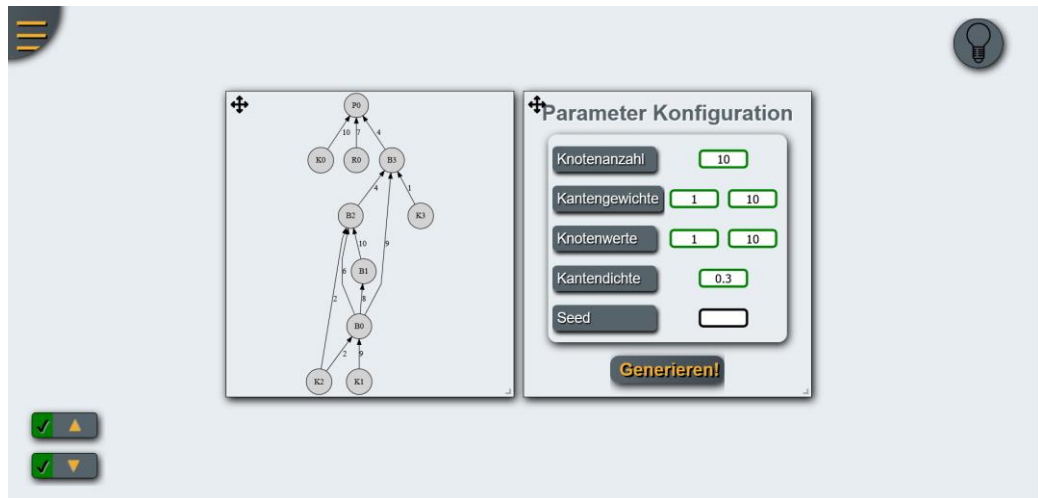


Abbildung A13: Parametrisierung der Aufgabe (Quelle: Eigene Darstellung)

Zuerst parametrisieren die Studierenden die Aufgabe. An dieser Stelle wird auch die Schwierigkeit der Aufgabe festgelegt.

2.

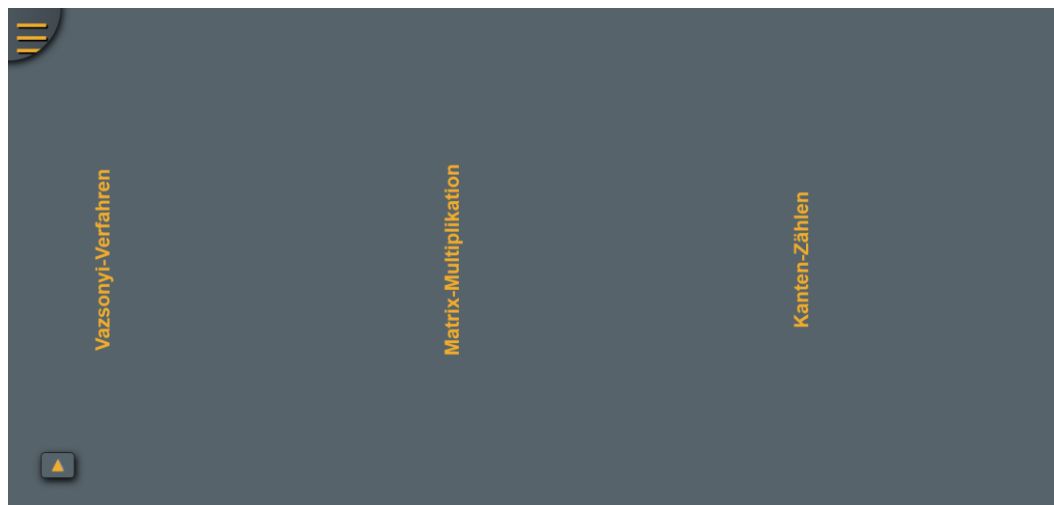


Abbildung A14: Wahl zwischen verschiedenen Lösungsvarianten (Quelle: Eigene Darstellung)

An dieser Stelle gibt es nun die Möglichkeit zwischen drei verschiedenen Lösungsvarianten zu wählen.

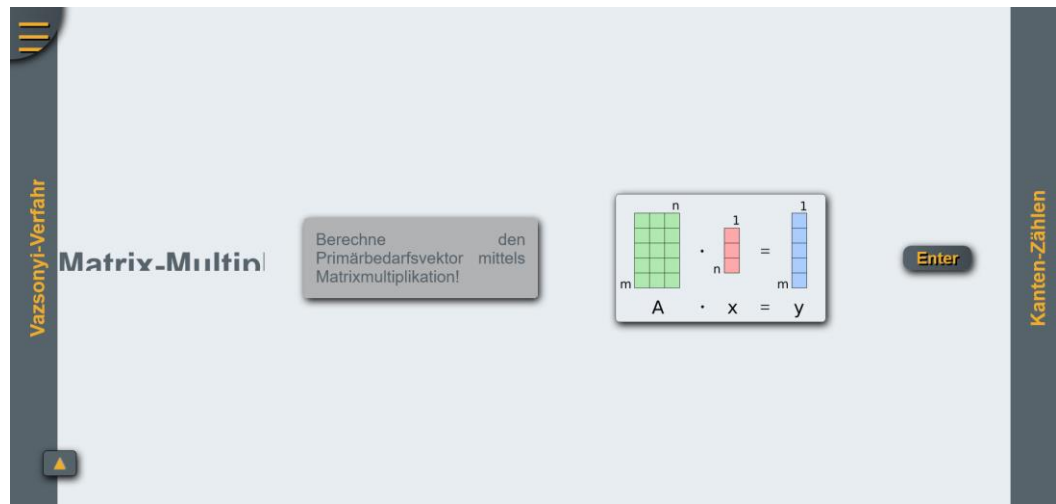


Abbildung A15: Hinweise zum gewählten Lösungsweg (Quelle: Eigene Darstellung)

Nach der Auswahl des Lösungsweges werden dem Nutzer, Hinweise zum entsprechenden Lösungsweg dargestellt.

3.

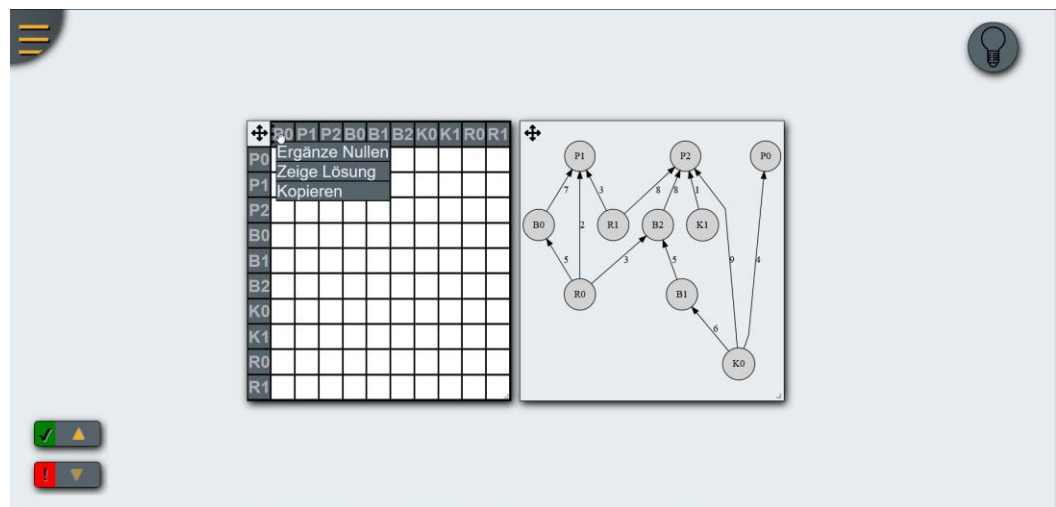


Abbildung A16: Ablesen der Adjazenzmatrix (Quelle: Eigene Darstellung)

In der folgenden Oberfläche muss der Nutzer die Adjazenzmatrix zu dem entsprechenden Gozintographen ablesen. Gezeigt werden außerdem mögliche (Hilfs-)Methoden, die bei dem Matricelement ausgeführt werden können.

4.

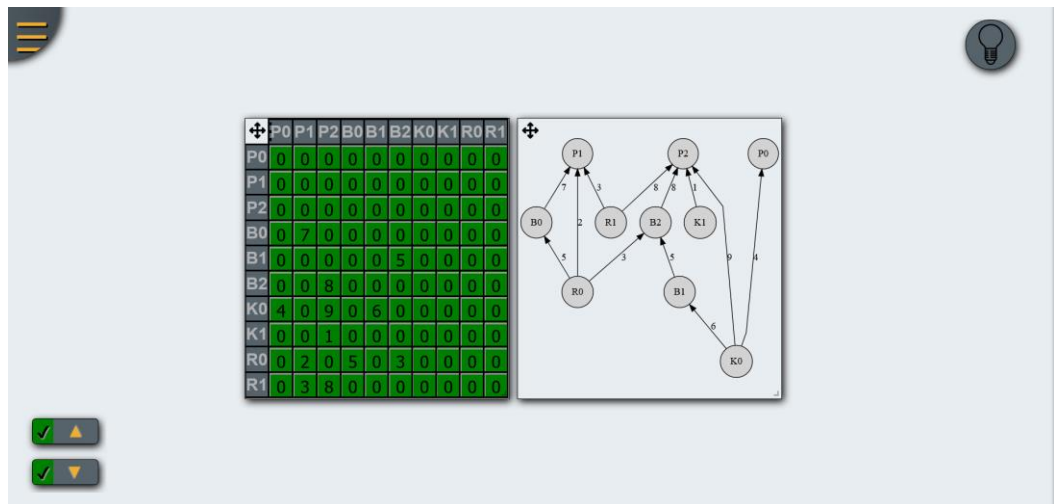


Abbildung A17: Vollständige und korrekte Adjazenzmatrix (Quelle: Eigene Darstellung)

Während des aktivierten Übungsmodus werden korrekte Eingaben des Users direkt grün hinterlegt angezeigt. Sobald die komplette Matrix ausgefüllt wurde, kann in die nächste Ansicht gewechselt werden.

5.

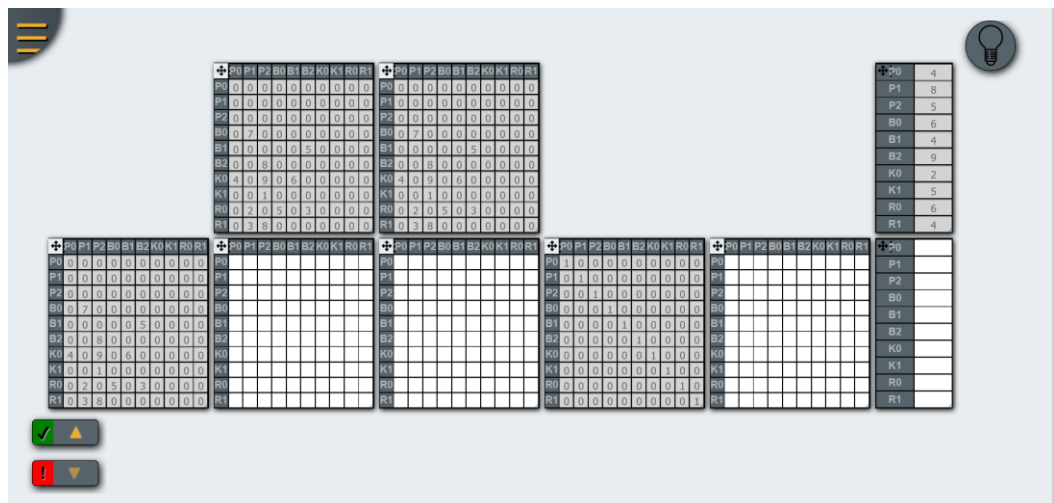


Abbildung A18: Matrixmultiplikation anhand der Stücklistenauflösung in ALADIN (Quelle: Eigene Darstellung)

Die Anzahl der Multiplikationsmatrizen wird dynamisch anhand des Gozintographen generiert.

13.5 Anforderungen an die Gamification in der Lernplattform ALADIN – grafisch

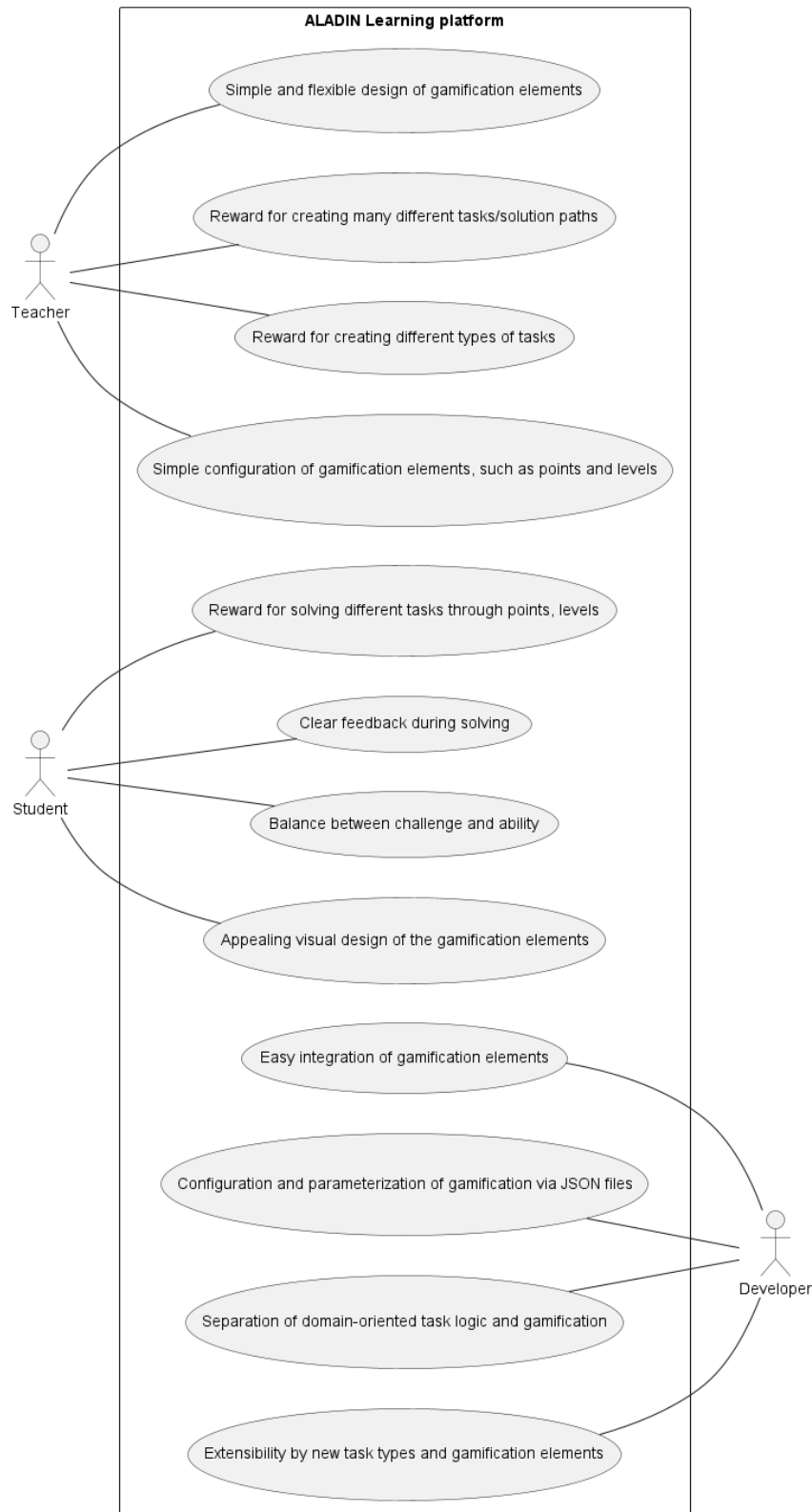


Abbildung A19: Gamification requirements (Quelle: Eigene Darstellung)

13.6 Verknüpfung der Aufzeichnung mit den Skills durch die Gamificationdefinition

Go2intograph.json	definition.ts	projects.json
<pre> 1 { 2 > "API": [...], 12 > "worker": {...}, 25 "UI": { 26 "currentTask": "go2intograph", 27 "taskNode": "strictPractice", 28 "taskData": {}, 29 "rootNode": 0, 30 "previousNode": 0, 31 > "topology": {...}, 48 > "edges": {...}, 93 "currentNode": 0, 94 "nodes": { 95 > "0": {"zoomScale": 1...}, 317 > "1": {"isValid": true...}, 338 > "2": {"zoomScale": 1...}, 477 > "3": {"zoomScale": 1...}, 758 > "4": {"zoomScale": 1...}, 961 > "5": {"zoomScale": 1...}, 1259 "6": { 1260 "zoomScale": 1, 1261 > "layouts": {...}, 1301 > "hints": {"active": false...}, 1308 "isValid": false, 1309 "isCorrect": false, 1310 "components": { 1311 "ogl": { 1312 "name": "Direktbedarfsmatrix", 1313 "type": "Matrix", 1314 "isValid": false, 1315 "isCorrect": false, 1316 "dependencies": { 1317 "data": "taskData_adjacencyMatrix" 1318 }, 1319 "methods": { 1320 "fillZeros": { 1321 "description": "Ergänze Nullen", 1322 "impact": 0.5 1323 }, 1324 "showsSolution": { </pre>	<pre> 21 { 22 "Go2intograph": { 23 "skillsDisplaySubject": "ProduktionswirtschaftslehreSubject", 24 "skillsMapping": { 25 "nodes__0_components__1__isValid": { 26 "value": true, 27 "skillsIds": ["AufgabeParametrisierenSkill"], 28 }, 29 "nodes__6_components__0__isValid": { 30 "value": true, 31 "skillsIds": ["AdjazenzmatrixAblesenSkill"], 32 }, 33 "nodes__7_components__(\d+)_isValid": { 34 "value": true, 35 "distinction": { 36 { 37 "skillsIds": ["GesamtbedarfsmatrixBestimmenSkill"], 38 "component": { "readOnly": false }, 39 "name": "Gesamtbedarfsmatrix", 40 }, 41 { 42 "skillsIds": ["MatrixMultiplikationsSkill", 43 "DirektbedarfsmatrixMultiplizierenSkill"], 44 "component": { "readOnly": false }, 45 "name": "Direktbedarfsmatrix", 46 }, 47], 48 }, 49 "nodes__7_components__3__isValid": { 50 "value": true, 51 "skillsIds": ["SekundaerbedarfsvektorBestimmenSkill"], 52 }, 53 }, 54 "Scheduling": { 55 "skillsDisplaySubject": "InformationsmanagementSubject", 56 "skillsMapping": { </pre>	<pre> 1 [2 { 3 "id": "ALADIN", 4 "subjects": [5 { 6 "id": "ProduktionswirtschaftslehreSubject", 7 "name": "Produktionswirtschaftslehre", 8 "iconClass": "", 9 "groups": [10 { 11 "id": "StuecklistenMatrixMultiplikationGroup", 12 "name": "Stuecklistenauflösung - Matrixmultiplikation", 13 "description": "" 14 }, 15], 16 "skills": [17 { 18 "id": "AdjazenzmatrixAblesenSkill", 19 "groupId": "StuecklistenMatrixMultiplikation", 20 "name": "Adjazenzmatrix ablesen", 21 "description": "", 22 "helpUrl": "", 23 "pointIncrement": 10, 24 "numPerformToCompletion": 25, 25 "pointIncrementInterval": 60, 26 "numMaxOccurrencesIncrementInterval": 5, 27 "cloneForUsedMethods": true 28 }, 29], 30 }, 31 { 32 "id": "StuecklistenMatrixMultiplikation", 33 "name": "StuecklistenMatrixmultiplikation", 34 "description": "", 35 "helpUrl": "", 36 "pointIncrement": 15, 37 "numPerformToCompletion": 50, 38 "pointIncrementInterval": 60, </pre>

Abbildung A20: Verbindung der Aufzeichnung mit den Skills - Gamificationdefinition (Quelle: Eigene Darstellung)

13.7 Test der Skill-Konfiguration

```
{
  "id": "ALADIN",
  "name": "ALADIN",
  "subjects": [
    {
      "id": "ProduktionswirtschaftslehreSubject",
      "name": "Produktionswirtschaftslehre",
      "iconClass": "",
      "groups": [
        {
          "id": "StuecklistenMatrixMultiplikationGroup",
          "name": "Stücklistenauflösung - Matrixmultiplikation",
          "description": ""
        }
      ],
      "skills": [
        {
          "id": "AdjazenzmatrixAblesenSkill",
          "groupId": "StuecklistenMatrixMultiplikationGroup",
          "name": "Adjazenzmatrix ablesen",
          "description": "",
          "helpurl": "",
          "pointIncrement": 10,
          "numPerformToCompletion": 25,
          "pointIncrementInterval": 60,
          "numMaxOccurrencesIncrementInterval": 5,
          "cloneForUsedMethods": true
        },
        {
          "id": "DirektbedarfsmatrixMultiplizierenSkill",
          "groupId": "StuecklistenMatrixMultiplikationGroup",
          "name": "Direktbedarfsmatrix multiplizieren",
          "description": "",
          "helpurl": "",
          "pointIncrement": 15,
          "numPerformToCompletion": 50,
          "numMaxOccurrencesIncrementInterval": 3
        }
      ]
    }
  ],
}
```

Abbildung A21: Definition des Moduls Produktionswirtschaftslehre (Quelle: Eigene Darstellung)

```
{
  "id": "ALADIN",
  "name": "ALADIN",
  "subjects": [
    {
      "id": "ProduktionswirtschaftslehreSubject",
      "name": "Produktionswirtschaftslehre",
      "iconClass": "",
      "groups": [
        {
          "id": "InformationsmanagementSubject",
          "name": "Informationsmanagement",
          "iconClass": "",
          "groups": [
            {
              "id": "MetraPotentialMethodesSkill",
              "name": "Metra-Potential-Methode durchführen",
              "description": "",
              "helpurl": "",
              "pointIncrement": 15,
              "numPerformToCompletion": 200,
              "pointIncrementInterval": 60,
              "numMaxOccurrencesIncrementInterval": 20
            },
            {
              "id": "ZeitanalyseDurchfuehrenSkill",
              "name": "Zeitanalyse durchführen",
              "description": "",
              "helpurl": "",
              "pointIncrement": 20,
              "numPerformToCompletion": 50,
              "pointIncrementInterval": 60,
              "numMaxOccurrencesIncrementInterval": 3
            }
          ]
        }
      ],
    }
  ],
}
```

Abbildung A22: Definition des Moduls Informationsmanagement (Quelle: Eigene Darstellung)

13.8 Test der Level-Konfiguration

```
"levels": {
  "1": {
    "percent": 2,
    "name": "Einsteiger",
    "iconClass": "fas fa-graduation-cap"
  },
  "2": {
    "percent": 4,
    "name": "Fortgeschrittener",
    "iconClass": "fas fa-graduation-cap"
  },
  "3": {
    "percent": 8,
    "name": "Kenner",
    "iconClass": "fas fa-graduation-cap"
  },
  "4": {
    "percent": 12,
    "name": "Experte",
    "iconClass": "fas fa-graduation-cap"
  },
  "5": {
    "percent": 21,
    "name": "Profi",
    "iconClass": "fas fa-graduation-cap"
  },
  "6": {
    "percent": 33,
    "name": "Spezialist",
    "iconClass": "fas fa-graduation-cap"
  },
  "7": {
    "percent": 49,
    "name": "Fachmann",
    "iconClass": "fas fa-graduation-cap"
  },
}
```

Abbildung A23: Definition der Level auf ALADIN-Ebene (Quelle: Eigene Darstellung)

```
"levels": {
  "1": {
    "percent": 3,
    "name": "Anfänger",
    "iconClass": "fas fa-graduation-cap"
  },
  "2": {
    "percent": 7,
    "name": "Neuling",
    "iconClass": "fas fa-graduation-cap"
  },
  "3": {
    "percent": 12,
    "name": "Einsteiger",
    "iconClass": "fas fa-graduation-cap"
  },
  "4": {
    "percent": 18,
    "name": "Schüler",
    "iconClass": "fas fa-graduation-cap"
  },
  "5": {
    "percent": 25,
    "name": "Lernender",
    "iconClass": "fas fa-graduation-cap"
  },
  "6": {
    "percent": 33,
    "name": "Fortschreitend",
    "iconClass": "fas fa-graduation-cap"
  },
  "7": {
    "percent": 42,
    "name": "Versierter",
    "iconClass": "fas fa-graduation-cap"
  },
}
```

Abbildung A24: Definition der Level auf Studienmodulebene (Quelle: Eigene Darstellung)

13.9 Test der Gamificationdefinition

```

"GoZintograph": {
  "skillsDisplaySubject": "ProduktionswirtschaftslehreSubject",
  "skillsMapping": {
    "nodes__0__components__1__isValid": {
      "value": true,
      "skillIds": ["AufgabeParametrisierenSkill"],
    },
    "nodes__6__components__0__isCorrect": {
      "value": true,
      "skillIds": ["AdjazenzmatrixAblesenSkill"],
    },
    "nodes__7__components__(\\d+)__isCorrect": {
      "value": true,
      "distinction": [
        {
          "skillIds": ["GesamtbedarfsmatrixBestimmenSkill"],
          "component": { "readOnly": false },
          "name": "Gesamtbedarfsmatrix",
        },
        {
          "skillIds": ["MatrixMultiplikationSkill", "DirektbedarfsmatrixMultiplizierenSkill"],
          "component": { "readOnly": false },
          "name": "Direktbedarfsmatrix",
        },
      ],
    },
    "nodes__7__components__3__isCorrect": {
      "value": true,
      "skillIds": ["SekundaerbedarfsvektorBestimmenSkill"],
    },
  },
},

```

Abbildung A25: Definition der Pfad-Skill Verknüpfung der Stücklistenauflösungsaufgabe (Quelle: Eigene Darstellung)

```

"Scheduling": {
  "skillsDisplaySubject": "InformationsmanagementSubject",
  "skillsMapping": {
    "nodes__0__components__0__isValid": {
      "value": true,
      "skillIds": ["AufgabeParametrisierenSkill"],
    },
    "nodes__2__components__0__isCorrect": {
      "value": true,
      "skillIds": ["MetraPotentialMethodesSkill"],
    },
    "nodes__3__components__0__isCorrect": {
      "value": true,
      "skillIds": ["ZeitanalyseDurchfuehrenSkill"],
    },
  },
},

```

Abbildung A26: Definition der Pfad-Skill Verknüpfung der Auftragsterminierungsaufgabe (Quelle: Eigene Darstellung)

13.10 Test der Stücklistenauflösung

Abbildung A27: Erfolgreiche Punktevergabe bei der Parametrisierung der Stücklistenauflösungsaufgabe (Quelle: Eigene Darstellung)

Abbildung A28: Keine Punktevergabe bei der Parametrisierung der Stücklistenauflösungsaufgabe, da Maximum der Lerneinheit erreicht (Quelle: Eigene Darstellung)

	P0	P1	B0	B1	B2	B3	B4	K0	K1	R0
P0										
P1										
B0	0	0	0	0	6	0	0	0	0	0
B1	7	0	0	0	0	0	0	0	0	0
B2	0	0	0	0	0	0	7	0	0	0
B3	0	2	0	0	0	0	0	0	0	0

Abbildung A29: Erfolgreiche Punktevergabe beim Ablesen der Adjazenzmatrix mit Hilfe (Quelle: Eigene Darstellung)

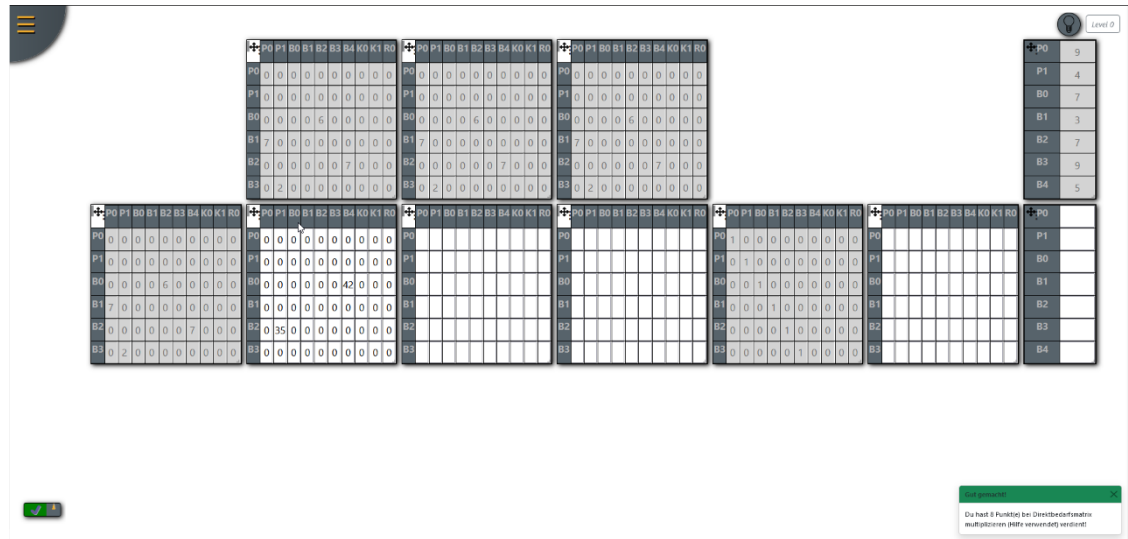


Abbildung A30: Erfolgreiche Punktevergabe bei der Direktbedarfsmatrixmultiplikation mit Hilfe (Quelle: Eigene Darstellung)

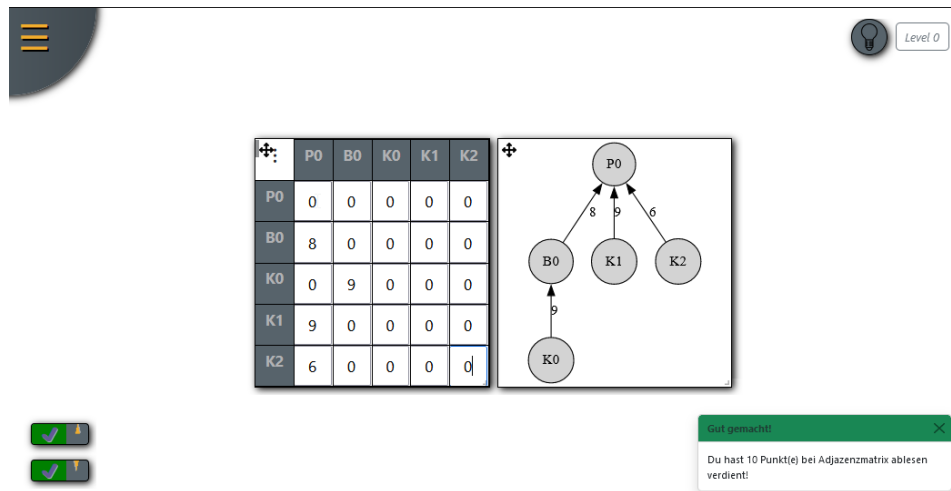


Abbildung A31: Erfolgreiche Punktevergabe beim Ablesen der Adjazenzmatrix (Quelle: Eigene Darstellung)

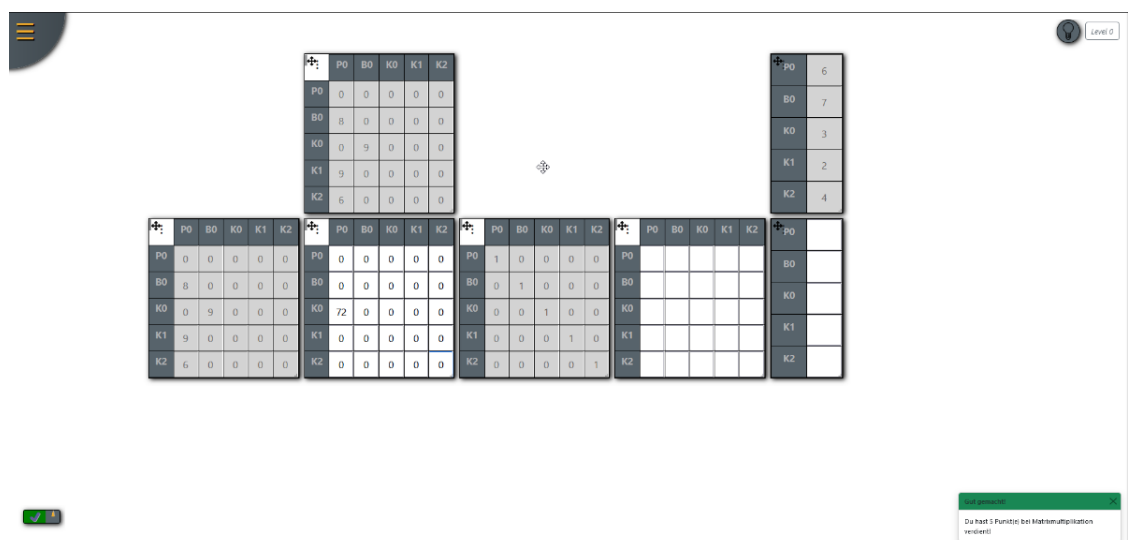


Abbildung A32: Erfolgreiche Punktevergabe bei der Direktbedarfsmatrixmultiplikation (Quelle: Eigene Darstellung)

13.11 Test der Auftragsterminierung

Parameter

Knotenanzahl: 10

Dauer: 15

Seed: 0

Generieren!

Vorgangsliste

Index	Vorgang	Dauer	Vorgänger
0	A	8	
1	B	4	A
2	C	12	A
3	D	9	C
4	E	12	C
5	F	7	E
6	G	8	D
7	H	1	B
8	I	15	D
9	J	14	F, G, H, I

Du hast 1 Punkte bei Aufgaben parametrisieren verdient!

Abbildung A33: Erfolgreiche Punktevergabe bei der Parametrisierung der Auftragsterminierungsaufgabe (Quelle: Eigene Darstellung)

Vorgangsliste

Index	Vorgang	Dauer	Vorgänger
0	A	8	
1	B	4	A
2	C	12	A
3	D	9	C
4	E	12	C
5	F	7	E
6	G	8	D
7	H	1	B
8	I	15	D
9	J	14	F, G, H, I

Netzplan-Diagramm mit Aktivitäten A-J und ihren Abhängigkeiten.

Du hast 10 Punkte bei Metra-Potenzial-Methode durchgeführt verdient!

Abbildung A34: Erfolgreiche Punktevergabe bei der Durchführung der Metra-Potenzial-Methode (Quelle: Eigene Darstellung)

Netzplan-Diagramm mit numerischen Werten in den Knoten und auf den Kanten.

Du hast 20 Punkte bei Zeitanalyse durchgeführt verdient!

Abbildung A35: Erfolgreiche Punktevergabe beim Durchführen der Zeitanalyse (Quelle: Eigene Darstellung)

13.12 Ergebnisse der Tests

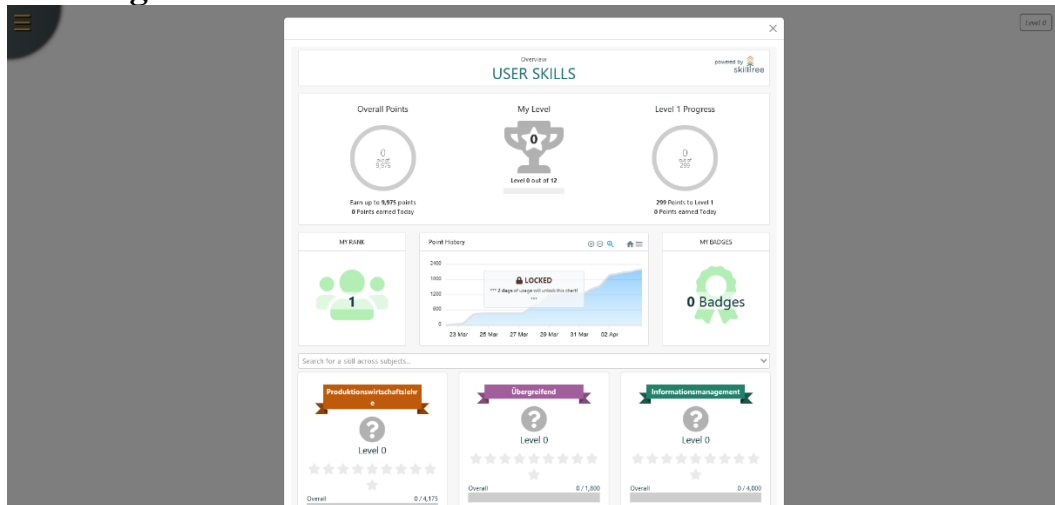


Abbildung A36: Initiale Skill-Ansicht zu Beginn des Tests (Quelle: Eigene Darstellung)

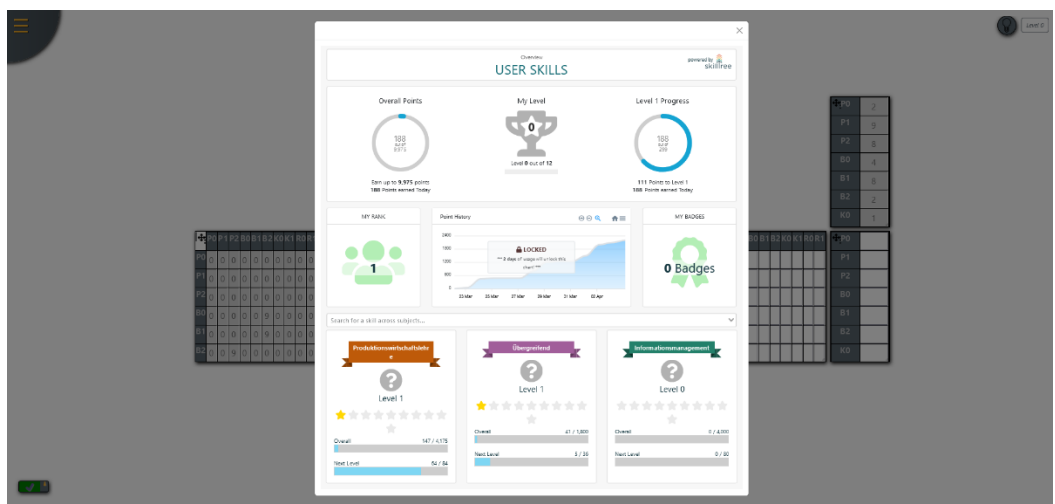


Abbildung A37: Finale Übersicht aller Punkte und Studienmodule nach dem Test (Quelle: Eigene Darstellung)

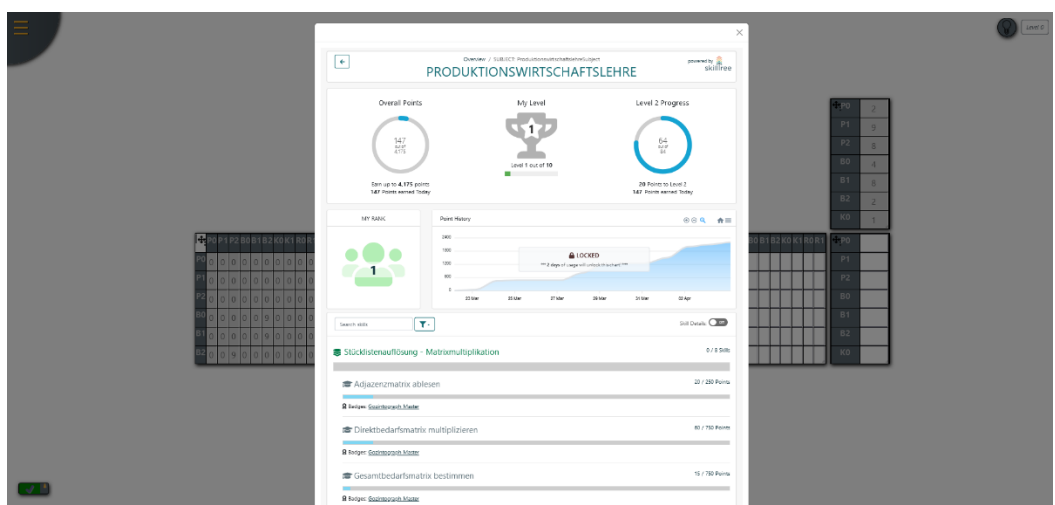


Abbildung A38: Finale Übersicht aller Punkte der Produktionswirtschaftslehre nach dem Test (Quelle: Eigene Darstellung)