



OST
Ostschweizer
Fachhochschule

12.06.2026 – FS 2026

Interaktives Pausenspiel der SCRJ

Bachelorarbeit

Projektteam: Jeriel Frei
Simeon Rhyner

Projektbetreuer: Frieder Loch

Praxispartner: Lakers Sport AG

Inhaltsverzeichnis

I Vorwort 1

1 Abstract 2

2 Management Summary 3

3 Danksagung 6

II Projektdokumentation 7

4 Einleitung 8
4.1 Problemstellung 8
4.2 Erwartetes Ergebnis 8
4.3 Forschungsfragen 8
4.4 Abgrenzung des Projektes 9
4.5 Vorgehen und Projektablauf 9

5 Recherche 10
5.1 Bestehende Pausenunterhaltungen bei anderen Clubs 10
5.2 Spiele mit vielen Teilnehmenden 11

6 Analyse 12
6.1 Anforderungen der Lakers 12
6.2 Spielideen 12
6.3 Bewertungskriterien und Bewertung 25
6.4 Mitmachwert 36

7 Konzept 38
7.1 Anwendungsfälle 38
7.2 Nichtfunktionale Anforderungen 45
7.3 Gemeinsame Spielauswertung 51
7.4 Mobile Teilnehmendenansicht 52
7.5 Administrationsbereich 52
7.6 Mockups 54

8 Systemkonzept und Architektur 59
8.1 Ziel der Umsetzung 59
8.2 Übersicht der verwendeten Technologien 60
8.3 Systemarchitektur 63
8.4 Datenmodell und Persistenz 68
8.5 Schnittstellen 74
8.6 Sicherheitsaspekte 76

9 Realisierung und Betrieb 79
9.1 Bedienoberfläche 79
9.2 Herausforderungen und Lösungsansätze 84
9.3 Deployment und Ausführung 86
9.4 Testen und Qualitätssicherung 92
9.5 Zusammenfassung 94

10 Evaluation 95
10.1 Evaluationsziel und Vorgehen 95
10.2 Anforderungsvalidierung 95
10.3 API-Lasttest 102
10.4 Nutzungsbezogene Evaluation 105
10.5 Gesamtbewertung und Limitationen 108
11 Schlussfolgerungen und Ausblick 109
11.1 Fazit der Arbeit 109
11.2 Beantwortung der Forschungsfragen 110
11.3 Kritische Würdigung 111
11.4 Ausblick 111
12 Glossary und Abkürzungen 112
13 Abbildungsverzeichnis 115
14 Tabellenverzeichnis 122
15 Literaturverzeichnis 124
16 Hilfsmittelverzeichnis 126
Anhang A: Technische Artefakte 127

Part I

Vorwort

1 Abstract

Diese Bachelorarbeit befasst sich mit der Entwicklung eines interaktiven Pausenspiels für die SCRJ Lakers. Ziel war es, ein digitales Spielkonzept zu entwerfen und als funktionsfähigen MVP umzusetzen, das während einem kleinen Zeitfenster in der zweiten Drittelpause im Stadion eingesetzt werden kann. Die Anwendung soll über mobile Geräte zugänglich sein, Inhalte auf dem Videowürfel darstellen und die Spielleitung bei Vorbereitung, Durchführung und Auswertung unterstützen.

Auf Basis der Anforderungen der Lakers Sport AG wurden verschiedene Spielideen analysiert und anhand organisatorischer, strategischer und technischer Kriterien bewertet. Darauf aufbauend wurde eine modulare Lösung mit den Spielmodi Quiz, Kreuzworträtsel und Bilderrätsel konzipiert. Die Umsetzung umfasst einen Administrationsbereich, eine mobile Teilnehmendenansicht, eine Präsentationsansicht für den Videowürfel, ein Backend mit zentraler Spielsteuerung sowie eine persistente Datenhaltung.

Die Evaluation zeigt, dass der entwickelte MVP die zentralen Abläufe eines interaktiven Stadion-Spiels abbildet. Vorbereitung, Teilnahme, Spielsteuerung, Antwortverarbeitung, Auswertung und Ermittlung der gewinnenden Person wurden umgesetzt und getestet. API-Lasttests bestätigten die Eignung der minimal dimensionierten Infrastruktur für den erwarteten Einsatzbereich und zeigten, dass verdoppelte AWS-Ressourcen auch deutlich höhere Lasten stabil abfangen. Der geplante Test auf dem realen Videowürfel konnte aufgrund Bauarbeiten im Stadion nicht durchgeführt werden und wurde durch eine Prüfung auf einem Display mit gleicher Auflösung ersetzt.

Die Arbeit zeigt, dass ein digitales Pausenspiel für die SCRJ Lakers technisch realisierbar ist. Für eine produktive Weiterführung sind insbesondere ein Test unter realen Stadionbedingungen, die Integration in die Lakers-App sowie Erweiterungen bei Betrieb, Rollenmodell und Scoreboard relevant.

2 Management Summary

Die Lakers Sport AG möchte die Pausenunterhaltung im Stadion stärker interaktiv gestalten. Im Mittelpunkt dieser Bachelorarbeit stand deshalb die Frage, wie ein digitales Pausenspiel umgesetzt werden kann, das während einem kurzen Zeitfenster in der zweiten Drittelpause verständlich, zuverlässig und mit der bestehenden Stadioninfrastruktur nutzbar ist. Das Spiel soll ein breites Publikum ansprechen, über mobile Geräte zugänglich sein, auf dem Videowürfel sichtbar inszeniert werden und gleichzeitig Möglichkeiten für Sponsorenintegration bieten.

Ausgehend von diesen Anforderungen wurden verschiedene Spielideen analysiert und bewertet. Wichtige Kriterien waren Verständlichkeit, Zielgruppeninklusion, technische Machbarkeit, Entwicklungsaufwand, Skalierbarkeit der Teilnahme, Sponsorenintegration und die Möglichkeit, gewinnende Personen zu bestimmen. Auf dieser Basis entstand das Konzept eines modularen interaktiven Spiels, das mehrere Spielmodi unterstützt und den Ablauf von der Vorbereitung bis zur Auswertung digital abbildet.

Als Ergebnis wurde ein funktionsfähiger MVP entwickelt. Die Lösung umfasst einen Administrationsbereich für die Vorbereitung und operative Steuerung, eine mobile Teilnehmendenansicht, eine Präsentationsansicht für den Videowürfel, ein Backend zur zentralen Spielsteuerung sowie ein Deployment auf AWS. Umgesetzt wurden die drei Spielmodi Quiz, Kreuzworträtsel und Bilderrätsel. Sie unterscheiden sich in Inhalt und Darstellung, verwenden aber denselben Grundablauf für Spielstart, Phasensteuerung, Antwortverarbeitung, Auswertung und Ermittlung der gewinnenden Person.

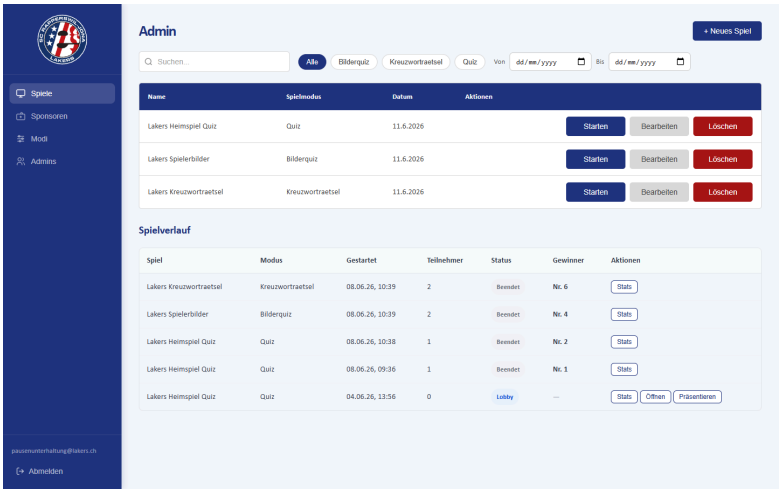


Abbildung 1: Zentrale Ansichten des MVP: Administrationsbereich
Quelle: Eigene Aufnahme

Die technische Architektur trennt öffentliche Ansichten, Administration und serverseitige Spiellogik klar voneinander. Das Backend verwaltet den massgeblichen Zustand einer Spielrunde und verteilt Änderungen gleichzeitig an Administrationsbereich, mobile Teilnehmendenansicht und Präsentationsansicht. Dadurch bleiben Spielleitung, Teilnehmende und Videowürfel während einer laufenden Runde synchron. Die Persistenz erfolgt über PostgreSQL, Medien werden in S3-kompatiblem Speicher abgelegt, und die Administrationsfunktionen sind durch eine Anmeldung geschützt.

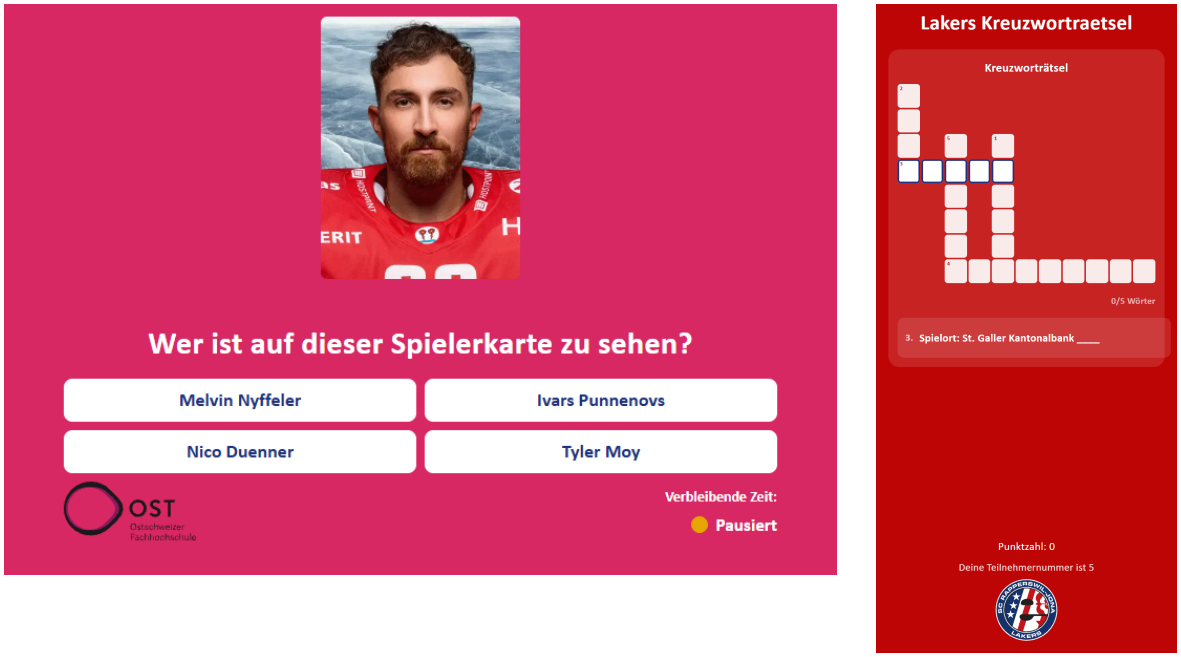


Abbildung 2: Zentrale Ansichten des MVP: Präsentationsansicht und mobile Teilnehmendenansicht
Quelle: Eigene Aufnahme

Die Evaluation zeigt, dass der MVP die definierten Kernabläufe für den untersuchten Einsatzbereich abdeckt. Die zentralen funktionalen Anforderungen für Vorbereitung, Teilnahme, Spielsteuerung, Antwortabgabe, Auswertung und Gewinnabwicklung wurden umgesetzt. Die wichtigsten nichtfunktionalen Anforderungen wurden durch Funktionstests, automatisierte Tests, manuelle Tests und API-Lasttests geprüft. Der API-Lasttest mit minimalen AWS-Ressourcen zeigte bei 100 gleichzeitigen Benutzenden eine durchschnittliche Antwortzeit von rund 24 ms und ein 95%-Perzentil von rund 68 ms. Bei 500 und insbesondere 1000 gleichzeitigen Benutzenden stiegen die Antwortzeiten in der Minimalvariante deutlich an. Der wiederholte Lasttest mit verdoppelten AWS-Ressourcen reduzierte das 95%-Perzentil bei 1000 Benutzenden auf rund 223 ms und bestätigt damit die Wirksamkeit der vorgesehenen Skalierung.

Eine wesentliche Limitation der Arbeit ist der fehlende vollständige Test unter realen Stadionbedingungen. Der geplante Test auf dem Videowürfel konnte wegen Bauarbeiten im Lakers-Stadion nicht durchgeführt werden, da der Videowürfel vom Strom getrennt war. Die Präsentationsansicht wurde deshalb auf einem Computerdisplay mit gleicher Auflösung geprüft. Damit konnte die technische Darstellung validiert werden, reale Faktoren wie Distanz, Lichtverhältnisse, Geräuschpegel, Ablenkung und Netzqualität konnten jedoch nur eingeschränkt beurteilt werden.

Für eine produktive Weiterführung sind mehrere Ausbauschritte relevant. Priorität hat ein Test während eines realen Stadionbetriebs, um Ablauf, Sichtbarkeit, Netzverhalten und Publikumsinteraktion gesamthaft zu prüfen. Zudem sollte die Integration in die Lakers-App vertieft werden, da sie Einstieg, Reichweite und Identifikation der Teilnehmenden verbessern kann. Weitere Erweiterungspunkte sind ein fein abgestuftes Rollenmodell für den Administrationsbereich, Monitoring für den Betrieb, ein saisonübergreifendes Scoreboard und zusätzliche Spielmodi

Die Arbeit zeigt, dass ein digitales Pausenspiel für die SCRJ Lakers technisch realisierbar ist und mit einem überschaubaren MVP einen vollständigen Spielablauf abdecken kann. Sie liefert damit eine belastbare Grundlage, auf der die Lakers Sport AG eine produktive Stadionlösung weiterentwickeln und unter realen Bedingungen einsetzen kann.

3 Danksagung

An dieser Stelle möchten wir allen Personen danken, die uns während der Bachelorarbeit unterstützt und begleitet haben.

Ein besonderer Dank gilt unserem Betreuer Prof. Dr. Frieder Loch für die fachliche Begleitung, die regelmässigen Rückmeldungen und die konstruktiven Hinweise während der gesamten Projektlaufzeit. Seine Unterstützung half uns dabei, fachliche Entscheidungen zu reflektieren, den Fokus der Arbeit zu schärfen und die Umsetzung gezielt weiterzuentwickeln.

Wir danken der Lakers Sport AG für die Möglichkeit, diese Bachelorarbeit in einem praxisnahen Umfeld durchführen zu dürfen. Der Einblick in die Anforderungen des Stadionbetriebs und die Gespräche zu Ablauf, Zielgruppe und technischer Infrastruktur waren für die Konzeption des interaktiven Pausenspiels sehr wertvoll. Unser Dank gilt insbesondere allen beteiligten Personen, die sich Zeit für Besprechungen, Rückfragen und Feedback genommen haben.

Ebenfalls danken wir allen Personen, die an Tests, Gesprächen und Rückmeldungen beteiligt waren. Ihre Einschätzungen haben dazu beigetragen, die Verständlichkeit, Bedienbarkeit und Praxistauglichkeit der Lösung besser einzuordnen.

Abschliessend möchten wir unseren Familien, Freunden und Mitstudierenden danken. Sie haben uns während der intensiven Arbeitsphase unterstützt, motiviert und durch fachlichen sowie persönlichen Austausch zum Gelingen dieser Arbeit beigetragen.

Rapperswil, Juni 2026
Jeriel Frei und Simeon Rhyner

Part II

Projektdokumentation

4 Einleitung

Ausgehend von der Aufgabenstellung der Lakers Sport AG untersucht diese Arbeit, wie die Pausenunterhaltung im Stadion stärker interaktiv gestaltet und das Publikum während der Drittelpausen aktiver in das Stadionerlebnis eingebunden werden können. Während eines Eishockeyspiels entsteht zu Beginn der Drittelpausen ein kurzer Zeitraum, in dem Zuschauende unterhalten werden können, ohne den regulären Spielbetrieb zu stören. Klassische Formen der Pausenunterhaltung sind dabei oft passiv geprägt, etwa durch Moderation, Animationen auf dem Videowürfel oder einzelne Aktionen auf dem Eis. Digitale Interaktionsformen bieten die Möglichkeit, eine grössere Anzahl Zuschauender gleichzeitig einzubeziehen und die Aufmerksamkeit im Stadion gezielt auf ein gemeinsames Erlebnis zu richten.

4.1 Problemstellung

Die zentrale Problemstellung dieser Arbeit liegt darin, ein digitales Pausenspiel so zu gestalten, dass es unter den konkreten Rahmenbedingungen eines Stadionbetriebs funktioniert. Das Spiel muss in wenigen Minuten verständlich sein, über mobile Geräte zugänglich sein, auf dem Videowürfel klar dargestellt werden und durch eine Spielleitung zuverlässig vorbereitet, gestartet, gesteuert und ausgewertet werden können. Gleichzeitig soll die Lösung für ein breites Publikum geeignet sein und Möglichkeiten zur Sponsorenintegration bieten. Diese Anforderungen führen zu einem Spannungsfeld zwischen einfacher Bedienung, technischer Machbarkeit, organisatorischer Durchführbarkeit und attraktivem Spielwert.

4.2 Erwartetes Ergebnis

Ziel dieser Bachelorarbeit war es deshalb, ein geeignetes Konzept für ein interaktives Pausenspiel der SCRJ Lakers zu erarbeiten, als funktionsfähigen MVP umzusetzen und unter den gegebenen Rahmenbedingungen zu evaluieren. Das erwartete Ergebnis war eine vollständige Anwendung mit mobiler Teilnehmendenansicht, Präsentationsansicht für den Videowürfel, Administrationsbereich, Backend, Datenhaltung und Deployment.

4.3 Forschungsfragen

Aus dieser Zielsetzung ergeben sich mehrere leitende Forschungsfragen. Erstens wird untersucht, welches Spielkonzept sich für ein interaktives Pausenspiel im Stadion eignet, wie dieses anhand strategischer, organisatorischer und technischer Kriterien ausgewählt werden kann und wie die Interaktion zwischen Zuschauenden, Spielleitung und Videowürfel gestaltet sein muss, damit das Spiel in kurzer Zeit verständlich und durchführbar bleibt. Zweitens wird betrachtet, ob ein MVP mit den gewählten Technologien die zentralen funktionalen und nichtfunktionalen Anforderungen ausreichend erfüllen kann. Schliesslich wird evaluiert, welche Grenzen unter den gegebenen Rahmenbedingungen bestehen und welche Schritte für eine produktive Weiterführung notwendig wären.

4.4 Abgrenzung des Projektes

Die Arbeit grenzt sich bewusst auf einen MVP ab. Eine vollständige produktive Integration in die bestehende Lakers-App, ein abschliessendes Rollen- und Berechtigungskonzept, ein saisonübergreifendes Scoreboard sowie ein umfassender Betrieb unter realen Stadionbedingungen sind nicht Bestandteil der finalen Umsetzung. Der geplante Test auf dem realen Videowürfel konnte aufgrund von Bauarbeiten im Stadion nicht durchgeführt werden und wurde durch eine Prüfung auf einem Display mit gleicher Auflösung ersetzt. Diese Einschränkung wird in der Evaluation berücksichtigt.

4.5 Vorgehen und Projektablauf

Die Arbeit wurde agil und iterativ durchgeführt. Für die Planung und Nachverfolgung der Aufgaben wurde in YouTrack ein Kanban-Board (Eska & Bichsel, 2021) mit den folgenden Sektionen: Backlog, Offen, In Bearbeitung, Wird geprüft, Mergen, Erledigt, Zurückgestellt und Abgebrochen geführt.

Durch regelmässige Besprechungen mit den beteiligten Personen konnten Zwischenergebnisse überprüft und das weitere Vorgehen angepasst werden. Die folgenden Phasen beschreiben deshalb die inhaltlichen Schwerpunkte der Arbeit und nicht einen strikt linearen Ablauf. Einzelne Aufgaben aus den Phasen wurden dabei nicht strikt nacheinander abgeschlossen, sondern je nach Erkenntnisstand erneut aufgegriffen.

Phase	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Recherche																		
Analyse																		
Konzept																		
Umsetzung																		
Evaluation																		

Abbildung 3: Zeitlicher Projektablauf nach Kalenderwochen

5 Recherche

5.1 Bestehende Pausenunterhaltungen bei anderen Clubs

Professionelle Sportveranstaltungen versuchen zunehmend, den Aufenthalt im Stadion über das eigentliche Spiel hinaus attraktiv zu gestalten. Insbesondere während Spielunterbrechungen, Drittelpausen oder Halbzeiten werden verschiedene Formen der Unterhaltung eingesetzt, um die Aufmerksamkeit der Zuschauenden aufrechtzuerhalten und das Stadionerlebnis zu verbessern. Ein attraktives Stadionerlebnis gilt als wichtiger Faktor für Fanbindung und Fan-Zufriedenheit. Studien zeigen, dass ein positives Fan-Erlebnis die Wahrscheinlichkeit erhöht, dass Zuschauende erneut Spiele besuchen und eine stärkere Bindung zum Verein entwickeln. (Giorgio et al., o. J.)

Zu den häufigsten Formen der Pausenunterhaltung zählen Animationen auf Videowänden oder Videowürfeln, Gewinnspiele, Quizformate oder moderierte Interaktionen mit dem Publikum. Viele Clubs setzen auch auf Unterhaltung durch Maskottchen, Tanz oder kurze Showeinlagen. Diese Aktivitäten sollen Wartezeiten überbrücken und gleichzeitig die Atmosphäre im Stadion aufrechterhalten. Die aktive Einbindung des Publikums wird dabei als zentraler Bestandteil der Fan-Experience betrachtet, da sie Zuschauende von passiven Beobachtenden zu aktiven Teilnehmenden macht.

Neben klassischen Unterhaltungselementen werden in modernen Stadien zunehmend digitale Technologien eingesetzt. Mobile Anwendungen oder interaktive Systeme ermöglichen es Zuschauenden beispielsweise, an Abstimmungen oder Spielen teilzunehmen und so direkt in das Veranstaltungsgeschehen eingebunden zu werden. Studien zeigen, dass solche technologiebasierten Interaktionsmöglichkeiten das Engagement und die Zufriedenheit der Fans steigern können. (Martins et al., 2023)

Insgesamt zeigt sich, dass Vereine verschiedene Strategien einsetzen, um Pausen aktiv zu gestalten und die Aufmerksamkeit der Fans während des gesamten Events zu halten. Besonders vielversprechend sind dabei Formate, die eine direkte Beteiligung eines grossen Teils des Publikums ermöglichen.

5.2 Spiele mit vielen Teilnehmenden

Spiele mit einer grossen Anzahl an Teilnehmenden stellen besondere Anforderungen ans technische Design und dessen Umsetzung. Im Kontext von Sportveranstaltungen können mehrere tausend Zuschauende gleichzeitig im Stadion anwesend sein. Interaktive Spiele müssen daher so konzipiert sein, dass sie für eine grosse Gruppe verständlich, leicht zugänglich und technisch skalierbar sind.

Ein zentraler Aspekt solcher Spiele ist die kollektive Interaktion. Beispiele hierfür sind Publikumsaktionen wie gemeinsame Bewegungen oder koordinierte Reaktionen wie die Mexican Wave. Berichte zeigen, dass solche kollektiven Aktionen durch das Verhalten weniger Initiatoren ausgelöst werden können und sich anschliessend selbstständig über grosse Publikumsgruppen ausbreiten. Diese Dynamik verdeutlicht, dass Spiele für grosse Menschenmengen häufig einfache Regeln benötigen, damit viele Personen spontan teilnehmen können. (Burchett, Michael H., 2022)

Mit der zunehmenden Digitalisierung von Stadien entstehen zudem neue Möglichkeiten für Massenteilnahme. Mobile Anwendungen ermöglichen es Fans, an Umfragen, Quizspielen oder Abstimmungen teilzunehmen. Untersuchungen zur digitalen Fan-Interaktion zeigen, dass solche Systeme tausende Teilnehmende gleichzeitig einbinden können und dadurch neue Formen der Beteiligung im Stadion schaffen (Martins et al., 2023). Auch neuere Konzepte wie Fan-Voting oder digitale Abstimmungen zeigen, dass grosse Gruppen erfolgreich in interaktive Prozesse eingebunden werden können. (Ante et al., 2024)

Ein weiteres Beispiel für digitale Massenteilnahme ist ein Fotoboard, bei dem Zuschauende eigene Fotos hochladen können, die anschliessend auf dem Videowürfel oder einer grossen Anzeige im Stadion dargestellt werden. Ein solches Format wurde von den Autoren an der IIHF Eishockey-Weltmeisterschaft 2026 in Zürich vor Ort beobachtet. Das Beispiel zeigt, dass auch einfache Beteiligungsformen ohne komplexe Spielmechanik einen grossen Teil des Publikums aktiv einbinden können.

Für Spiele mit vielen Teilnehmenden sind insbesondere folgende Eigenschaften entscheidend:

- Einfache Regeln, die ohne lange Erklärung verstanden werden
- Kurze Spieldauer, um in Spielpausen integriert werden zu können
- Skalierbarkeit, damit sowohl wenige als auch viele Zuschauende teilnehmen können
- Klare visuelle Darstellung, beispielsweise über grosse Videowände im Stadion

Solche Eigenschaften sind notwendig, um sicherzustellen, dass Interaktionen auch in einer Umgebung mit vielen gleichzeitig anwesenden Personen funktionieren. Daher konzentrieren sich viele stadionbasierte Spiele auf einfache Mechaniken wie Quizfragen, Abstimmungen oder Schätzspiele, die sich leicht auf grosse Gruppen übertragen lassen.

6 Analyse

6.1 Anforderungen der Lakers

In der ersten Phase des Projektes wurde ein Kickoff Meeting mit der Leiterin des Marketingteams der Lakers Sport AG durchgeführt. Unter anderem wurden an diesem Meeting die Anforderungen der Lakers an ein Spiel im Stadion geklärt. Als zentrale Anforderungen wurden eine breite Zielgruppe, eine verständliche Spielmechanik und die Umsetzbarkeit mit der bestehenden Stadioninfrastruktur festgehalten. Zusätzlich soll das Spiel eine Sponsorenintegration ermöglichen, damit ein Werbepartner sichtbar eingebunden und ein Gewinn für die Teilnehmenden bereitgestellt werden kann. Eine zentrale Rahmenbedingung ist das kurze Zeitfenster während der zweiten Drittelpause. Für das Pausenspiel stehen maximal vier Minuten zur Verfügung. Während des Spiels kann die Stadionbeleuchtung für die Darstellung auf dem Videowürfel reduziert werden. Für die anschliessende Eisaufbereitung muss die reguläre Beleuchtung jedoch wieder eingeschaltet sein, wodurch die verfügbare Spieldauer begrenzt ist.

6.2 Spielideen

In den folgenden Kapiteln werden die verschiedenen Spielideen vorgestellt und bewertet, welche für die Umsetzung infrage kommen. Dabei werden die Ideen in Bezug auf die Anforderungen der Lakers analysiert und anhand der definierten Kriterien bewertet, um eine begründete Auswahl für die Umsetzung zu treffen. Diese Ideen wurden in einem kreativen Prozess vor dem Kickoffmeeting entwickelt, der ein Brainstorming und Inspirationen aus bestehenden Spielen und Unterhaltungsformaten umfasst, welche in einer Mindmap festgehalten wurden. Die Mindmap zeigt die gesammelten Ansätze für mögliche Pausenspiele (siehe Abbildung 4).

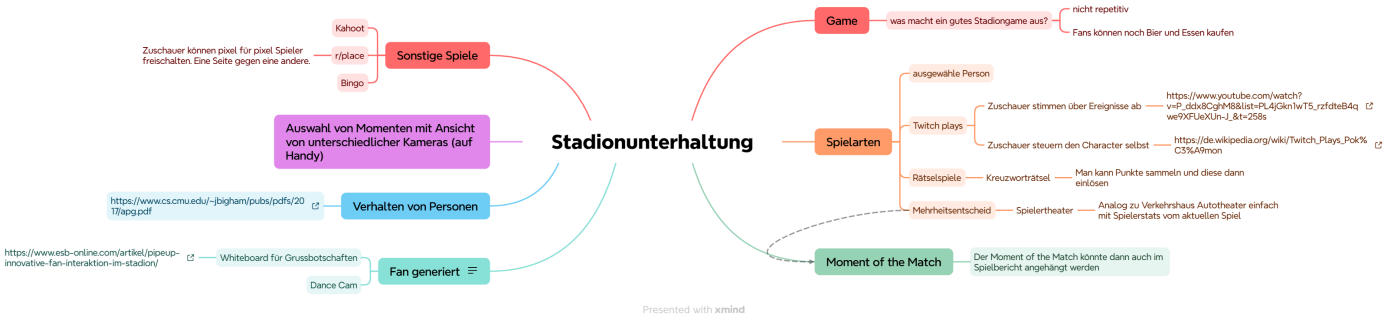


Abbildung 4: Mindmap
Quelle: Eigene Darstellung, erstellt mit Xmind

6.2.1 Stadion-Shootout

Beim Stadion-Shootout werden die Zuschauenden im Stadion über die Lakers-App in zwei Teams aufgeteilt, die in einem interaktiven Penalty-Schiessen gegeneinander antreten. Ein Team übernimmt die Rolle des Schützen, das andere Team die Rolle des Torwarts. Beide Teams stimmen in der App über eine von drei Richtungen ab: links, Mitte oder rechts. Das offensive Team legt damit fest, wohin geschossen wird. Das defensive Team entscheidet gleichzeitig, in welche Richtung sich der Torwart bewegt. Nach Ablauf der Abstimmung werden die Abstimmungsresultate beider Teams verglichen. Wählt das defensive Team dieselbe Richtung wie das offensive Team, gilt der Schuss als gehalten und das defensive Team gewinnt die Runde. Unterscheiden sich die Richtungen, erzielt das offensive Team ein Tor. Das Ergebnis wird anschliessend auf dem Videowürfel visualisiert. Die Smartphone-Sicht zeigt die Auswahlmöglichkeiten für die beiden Teams (siehe Abbildung 5).



Abbildung 5: Sketch Smartphone Sicht vom offensiven & defensiven Team
Quelle: Eigene Darstellung, erstellt mit Figma

Die Zuschauenden erleben dadurch direkt, wie sich die gesammelten Stimmen auf den Schuss und die defensive Reaktion auswirken. Die Visualisierung auf dem Videowürfel ist als gemeinsame Stadionansicht vorgesehen (siehe Abbildung 6).

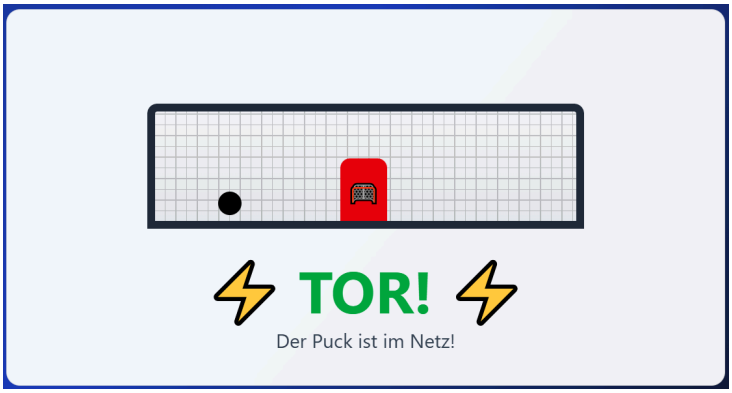


Abbildung 6: Sketch Videowürfel Sicht
Quelle: Eigene Darstellung, erstellt mit Figma

Spielablauf

Das Spiel besteht aus einer kurzen Anmeldephase, mehreren Abstimmungsrunden, der Visualisierung der Penaltys auf dem Videowürfel und einer abschliessenden Punktevergabe. In jeder Runde agieren die Teams abwechselnd in der offensiven und defensiven Rolle. Pro Runde wählen beide Teams über die App je eine Richtung aus den drei Optionen links, Mitte und rechts. Insgesamt werden drei Runden gespielt. Das Team mit den meisten erzielten Treffern gewinnt das Spiel.

Ranking / Gewinn

Das Team mit den meisten Punkten gewinnt das Stadion-Shootout. Um Fairness zu gewährleisten, wird der tatsächliche Gewinnpreis unter allen Mitgliedern des Siegerteams ausgelost.

Sponsorenintegration

Das Spiel bietet attraktive Integrationsmöglichkeiten für Sponsoren:

- Bereitstellung des Gewinns
- Branding in der App (z.B. „präsentiert von …“)
- Integration des Sponsorlogos auf dem Videowürfel
- Virtuelle Bandenwerbung hinter dem Tor auf dem Videowürfel
- Einblendung während der Abstimmungsphase

Durch die digitale Visualisierung auf dem Videowürfel und die Integration in die App wird eine hohe Sichtbarkeit gewährleistet.

Dauer

Die Gesamtdauer beträgt ca. 6.5 Minuten und gliedert sich wie folgt:

- Anmeldephase: 1 Minute
- Spielphase: 3 Runden à 1.5 Minuten (4.5 Minuten)
- Auswertungs- und Gewinnphase: 1 Minute

6.2.2 Stadion-Sprint

Das Publikum kann einen Eishockeyspieler (nachfolgend als „Spielfigur“ bezeichnet) auf dem Videowürfel mithilfe der Lautstärke im Stadion steuern. Das Ziel ist es, möglichst viele Hindernisse zu überwinden, um ins Ziel zu gelangen, ohne einen Unfall zu verursachen.

Spielablauf

Das Publikum kann durch das Erzeugen von Geräuschen (z.B. Klatschen, Rufen) die Spielfigur steuern. Je lauter die Geräusche, desto höher bewegt sich die Spielfigur auf dem Videowürfel. Die Spielfigur bewegt sich automatisch von links nach rechts durch eine Reihe von Hindernissen. Das Publikum muss die Höhe der Spielfigur so steuern, dass er nicht mit den Hindernissen kollidiert.

Ranking / Gewinn

Da das gesamte Stadion ohne Smartphone mitspielen kann, ist es nicht möglich, eine gewinnende Person zu bestimmen.

Sponsorenintegration

Das Spiel bietet attraktive Integrationsmöglichkeiten für Sponsoren:

- Integration des Sponsorlogos auf dem Videowürfel
- Custom-Design der Spielfigur und der Hindernisse im Stil des Sponsors

Dauer

Die Spieldauer kann mit dem Ziel flexibel gestaltet werden, somit kann eine maximale Spieldauer von 4 Minuten festgelegt werden. Eine untere Abgrenzung ist jedoch nicht gegeben.

Inspiration

Die Idee zu diesem Spiel kam vom bekannten Mobile Game „Flappy Bird“, das 2013 veröffentlicht wurde. Das Grundprinzip der seitlichen Bewegung durch Hindernisse ist im Originalspiel erkennbar (siehe Abbildung 7).

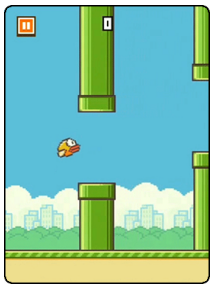


Abbildung 7: Screenshot originales Flappy Bird
Quelle: (DotGears, o. J.)

6.2.3 Stadion-Plays

Das Stadion wird in zwei Hälften geteilt. Jede Stadionhälfte steuert gemeinsam eine eigene Figur auf dem Videowürfel. Die Zuschauenden einer Seite geben ihre Steuerbefehle über die Lakers-App ab. Diese Eingaben werden gesammelt und zu einer gemeinsamen Bewegung der Figur zusammengeführt. Das Prinzip entspricht damit einer kollektiven Steuerung wie bei Twitch Plays Pokémon, übertragen auf zwei Teams im Stadion. Die Videowürfel-Sicht zeigt die gemeinsame Spielfläche für das Publikum (siehe Abbildung 8).

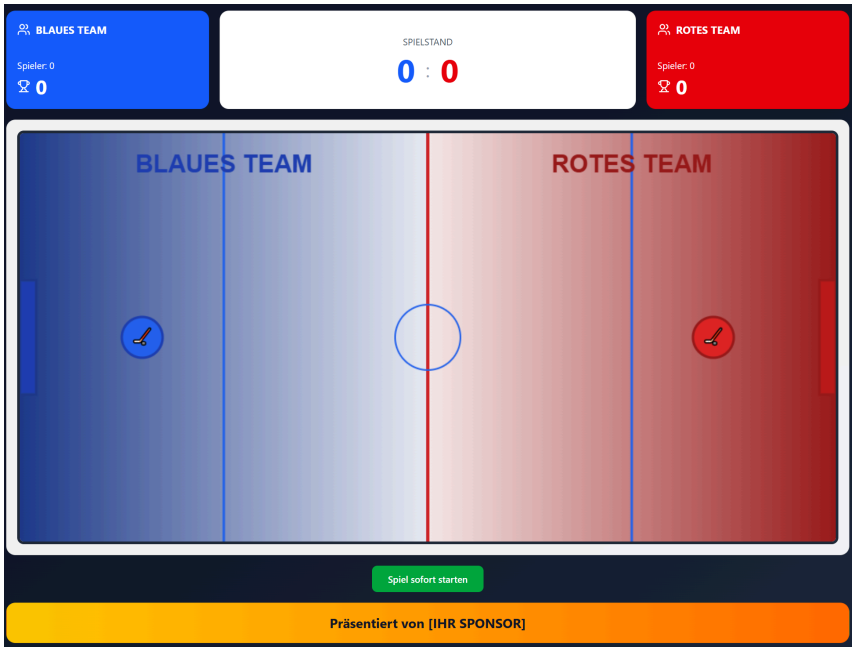


Abbildung 8: Sketch Sicht Videowürfel
Quelle: Eigene Darstellung, erstellt mit Figma

Spielablauf

Die Zuschauenden können sich über die Lakers-App für StadionPlays anmelden und sich für eine der beiden Seiten entscheiden. Nach der Anmeldung wird das Spiel gestartet. Die Zuschauenden steuern nicht je eine eigene Figur, sondern beeinflussen gemeinsam die Figur ihrer Seite. Jede Eingabe wird als Vorschlag für die nächste Bewegung verstanden. Aus den gesammelten Eingaben entsteht ein gemeinsamer Steuerbefehl, mit dem die Figur Hindernissen ausweicht und durch das Eisfeld navigiert. Die Smartphone-Sicht zeigt die vorgesehenen Eingaben für die gemeinsame Steuerung (siehe Abbildung 9).

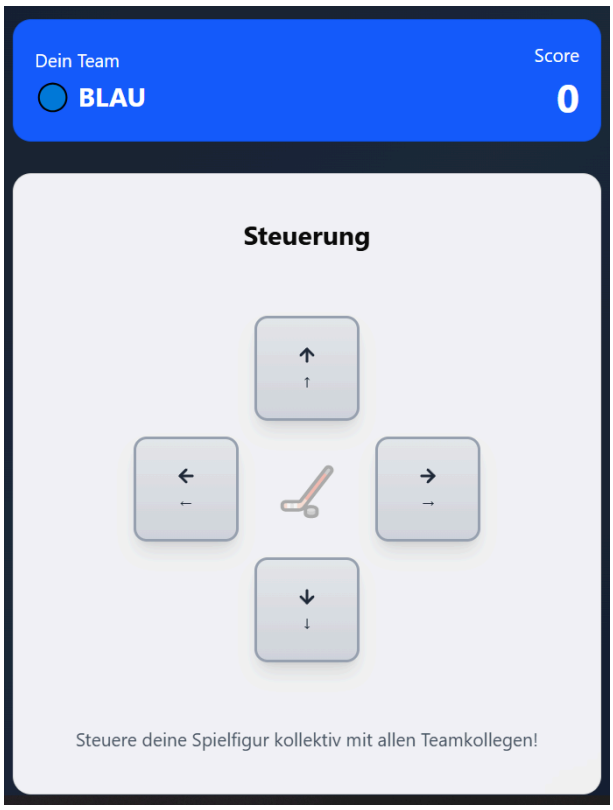


Abbildung 9: Sketch Sicht Smartphone
Quelle: Eigene Darstellung, erstellt mit Figma

Das Ziel ist es ein Tor beim gegnerischen Team zu erzielen. Es werden jeweils alle Inputs der Zuschauenden einer Seite ausgewertet und auf eine virtuelle Tastatur übertragen, welche die gemeinsame Spielfigur steuert.

Ranking / Gewinn

Bei diesem Spiel gibt es kein faires Ranking, da die Steuerung der Spielfigur von allen Zuschauenden übernommen wird. Es gibt eine Gewinnseite, bei der ein Gewinn verlost werden kann. Bei einem Gewinn wird die Person über die App benachrichtigt.

Sponsorenintegration

Das Spiel bietet attraktive Integrationsmöglichkeiten für Sponsoren:

- Branding in der App (z.B. „präsentiert von …“)
- Integration des Sponsorlogos auf dem Videowürfel
- Custom-Design der Spielfigur und der Eisfläche im Stil des Sponsors

Dauer

Die Gesamtdauer beträgt ca. 5 Minuten und gliedert sich wie folgt:

- Anmeldephase: 1 Minute
- Spielphase: 4 Minuten

Inspiration

Die Spielidee wurde von Twitch Plays Pokémon (Twitch Plays Pokémon, o. J.) inspiriert. Dabei handelt es sich um einen Twitch-Stream, bei dem Zuschauende über Chatbefehle gemeinsam ein Pokémon-Spiel steuern. Die einzelnen Eingaben werden gesammelt und als Steuerbefehle an das Spiel weitergegeben, wodurch eine grosse Gruppe gemeinsam Einfluss auf eine Spielfigur nehmen kann. Dieses Prinzip wurde auf den Stadionkontext übertragen: Statt Chatbefehlen würden die Zuschauenden ihre Eingaben über die Lakers-App abgeben, während die gemeinsame Spielfigur auf dem Videowürfel dargestellt wird.

6.2.4 Stadion-Quiz

Dieses Spiel besteht aus drei einzelnen Spielmodi.

- Quiz-Fragen
- Kreuzworträtsel
- Bilderrätsel

Das Marketing-Team kann sich für jedes Hockeyspiel zwischen diesen drei Spielmodi entscheiden, um die Zuschauenden zu unterhalten. Zudem kann das Marketing-Team auch entscheiden, welcher Inhalt in den Spielen verwendet wird. Zum Beispiel könnte das Marketing-Team entscheiden, dass die Fragen im Quiz sich um die Geschichte der Lakers drehen oder dass die Bilder im Bilderrätsel von den Spielerinnen und Spielern der Lakers stammen. In den folgenden Kapiteln werden die einzelnen Spielmodi genauer beschreiben.

Quiz-Fragen:

In diesem Spielmodus werden den Zuschauenden Fragen zu verschiedenen Themen rund um die Lakers gestellt. Die Zuschauenden können ihre Antwort über die Lakers-App eingeben. Nach Ablauf der Zeit werden die Antworten ausgewertet und die Punkte entsprechend vergeben. Bei jeder Frage gibt es vier Antwortmöglichkeiten, von denen nur eine korrekt ist. Die Zuschauenden haben 20 Sekunden Zeit, um ihre Antwort einzugeben. Die Darstellung auf dem Videowürfel zeigt Frage und Antwortmöglichkeiten (siehe Abbildung 10).

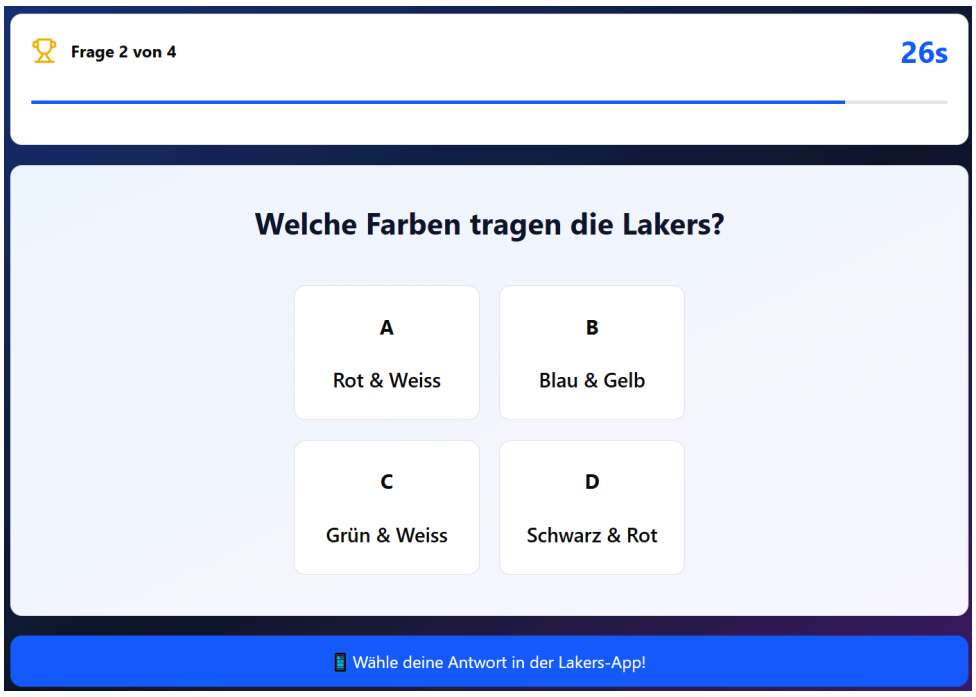


Abbildung 10: Sketch Videowürfel Sicht
Quelle: Eigene Darstellung, erstellt mit Figma

Kreuzworträtsel:

In diesem Spielmodus müssen die Zuschauenden ein Kreuzworträtsel lösen, das sich um die Lakers dreht. Das Kreuzworträtsel besteht aus 5 Fragen, die die Zuschauenden beantworten müssen, um das Rätsel zu lösen. Die Fragen können sich auf die Geschichte der Lakers, die Spielerinnen und Spieler, die Erfolge oder andere interessante Fakten beziehen. Die Zuschauenden haben 1.5 Minuten Zeit, um das Kreuzworträtsel zu lösen. Die Videowürfel-Sicht zeigt die Hinweise und das Rätselraster (siehe Abbildung 11).

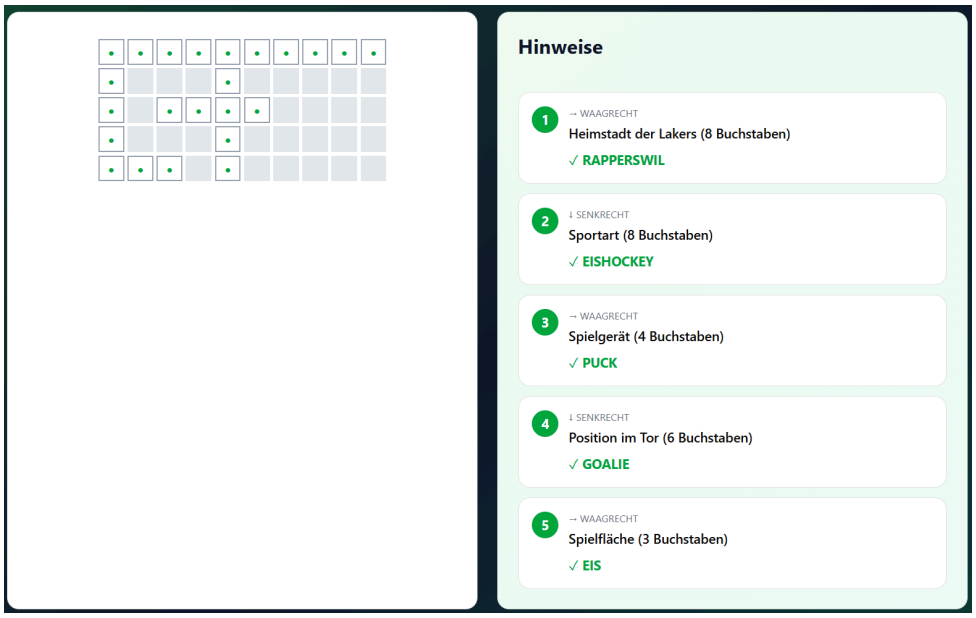


Abbildung 11: Sketch Kreuzworträtsel Videowürfel Sicht
Quelle: Eigene Darstellung, erstellt mit Figma

Bilderrätsel:

In diesem Spielmodus werden den Zuschauenden Bilder, die zum Beispiel verdreht sind, gezeigt, die mit den Lakers in Verbindung stehen. Die Zuschauenden müssen erraten, was auf den Bildern zu sehen ist. Die Bilder können von Spielerinnen und Spielern, Spielen oder anderen Aspekten der Lakers handeln. Die Zuschauenden haben 20 Sekunden Zeit, um ihre Antwort einzugeben. Das Spiel besteht aus 3 Bildern, die die Zuschauenden erraten müssen. Die geplante Darstellung des Bilderrätsels auf dem Videowürfel zeigt die reduzierte visuelle Frageansicht (siehe Abbildung 12).

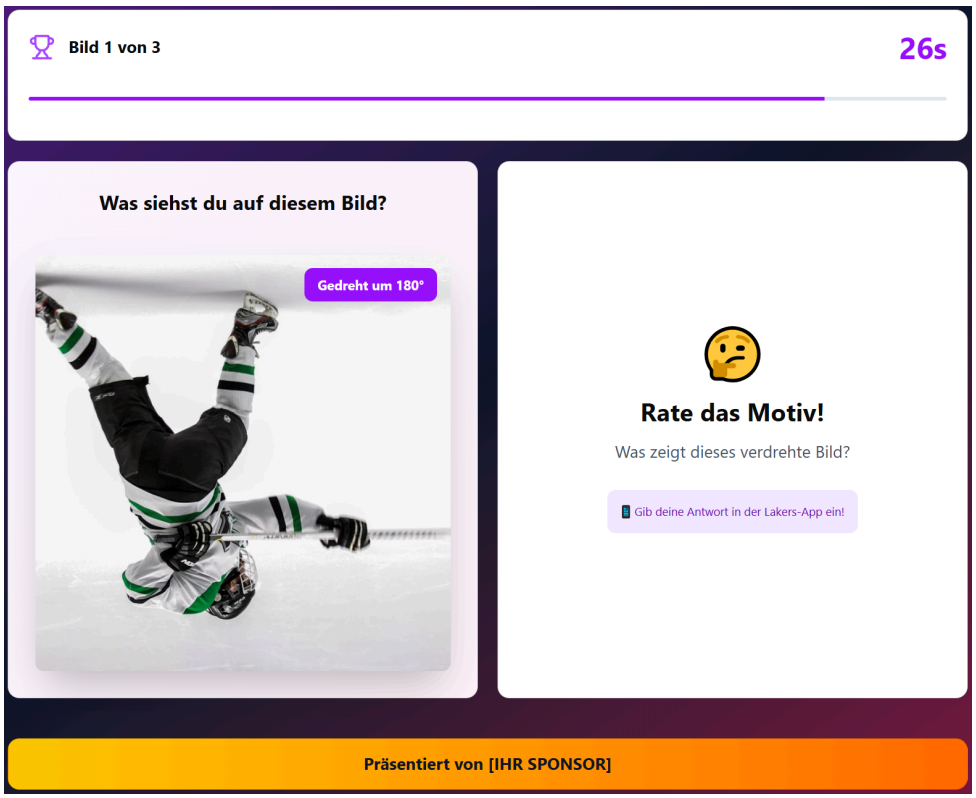


Abbildung 12: Sketch Bilderrätsel Videowürfel Sicht
Quelle: Eigene Darstellung, erstellt mit Figma

Spielablauf

Die Zuschauenden können sich über die Lakers-App für das Quiz anmelden. Nach der Anmeldung wird das Spiel gestartet. Dabei wird das Spiel auf dem Videowürfel visualisiert, damit alle Zuschauenden im Stadion das Spiel verfolgen können. Die Zuschauenden können ihre Antworten über die Lakers-App eingeben. Nach Ablauf der Zeit werden die Antworten ausgewertet und die Punkte entsprechend vergeben. Am Ende jeder Spielrunde wird die richtige Antwort auf dem Videowürfel angezeigt, damit die Zuschauenden sehen können, ob sie die Frage richtig beantwortet haben.

Ranking / Gewinn

Um die gewinnende Person zu ermitteln, gibt es zwei verschiedene Möglichkeiten:

1. Der Gewinn wird unter allen Teilnehmenden ausgelost, die die meisten Punkte erreicht haben.
2. Der Gewinn wird an die teilnehmende Person vergeben, die die meisten Punkte erreicht hat und die schnellste Antwortzeit hat.

Die Gewinnbenachrichtigung erfolgt über die Lakers-App. Die Preisübergabe findet direkt im Stadion statt.

Sponsorenintegration

Die Sponsorenintegration ist bei dieser Spielidee vor allem über Preis, App und Videowürfel möglich:

- Bereitstellung des Gewinnpreises
- Branding in der App (z.B. „präsentiert von …“)
- Integration des Sponsorlogos auf dem Videowürfel

Dauer

Die Gesamtdauer beträgt ca. 3 Minuten und gliedert sich wie folgt:

- Anmeldephase: 1 Minute
- Spielphase: 1.5 Minuten
- Auswertungs- und Gewinnphase: 30 Sekunden

6.2.5 Stadion-Duell

Die Teilnehmenden steuern eine Spielfigur auf einem Hockeyfeld, der versucht, einen Puck ins gegnerische Tor zu schiessen. Das Ganze wird als Turnier organisiert, bei dem immer die gewinnende Person weiterkommt. Die Smartphone-Sicht zeigt die vorgesehene Steuerung für das Duell (siehe Abbildung 13).

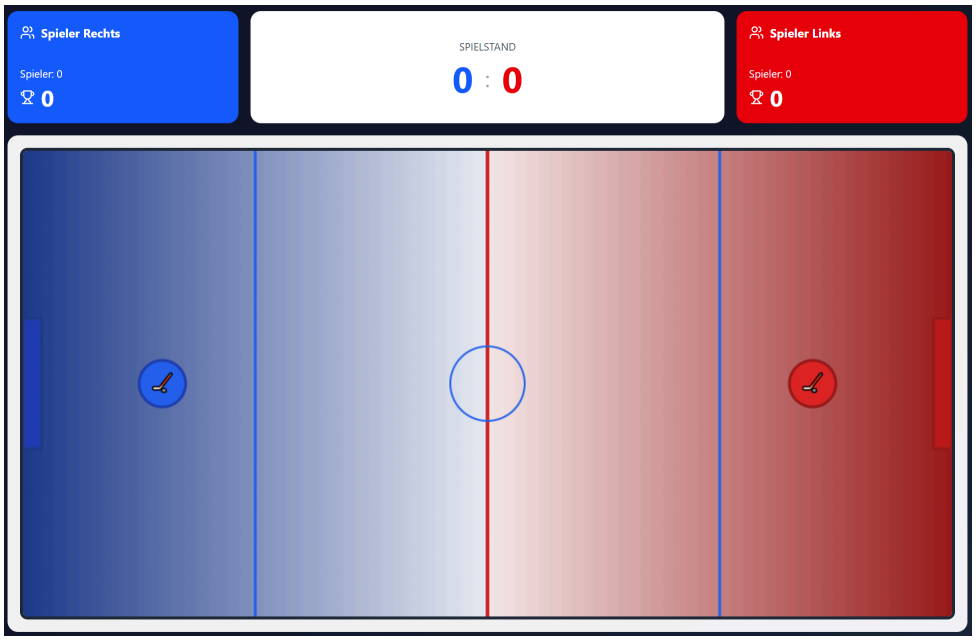


Abbildung 13: Sketch Smartphone Sicht
Quelle: Eigene Darstellung, erstellt mit Figma

Spielablauf

Zu Beginn melden sich die Zuschauenden über die Lakers-App für das Duell an. Anschliessend wird ein Turnierbaum erstellt, bei dem die Teilnehmenden in 1vs1-Duellen gegeneinander antreten. Eine Runde dauert 30 Sekunden und die teilnehmende Person mit den meisten Toren gewinnt die Runde und wird in der nächsten Runde mit einer anderen teilnehmenden Person fortgesetzt. Falls es nach 30 Sekunden unentschieden steht, gewinnt die teilnehmende Person, welcher die meisten Interaktionen mit dem Puck hatte.

Ranking / Gewinn

Die gewinnende Person hat das Finale für sich entschieden, weitere Platzierungen werden anhand des Turnierbaums ermittelt. Die Gewinnbenachrichtigung erfolgt über die Lakers-App.

Sponsorenintegration

Das Spiel bietet attraktive Integrationsmöglichkeiten für Sponsoren:

- Integration des Sponsorlogos auf dem Videowürfel
- Custom-Design der Spielfigur und der Eisfläche im Stil des Sponsors
- Branding in der App (z.B. „präsentiert von …“)

Dauer

Die Gesamtdauer beträgt ca. 5 Minuten und gliedert sich wie folgt:

- Anmeldephase: 1 Minute
- Spielphase: 4 Minuten bei 256 Teilnehmenden

6.3 Bewertungskriterien und Bewertung

Im folgenden Kapitel werden die Bewertungskriterien für die Evaluation der Spielideen definiert sowie die Bewertung der Spielideen anhand dieser Kriterien beschrieben. Die Bewertungskriterien wurden nach dem ersten Treffen mit den Lakers und der anschliessenden Analyse der Anforderungen der Lakers definiert. Jedes Bewertungskriterium wird mit einer kurzen Beschreibung und einer Gewichtung versehen, um den Einfluss des Kriteriums auf die Gesamtbewertung auszuweisen. Als Skala wird bei jedem Kriterium eine Skala von 1-3 Punkten verwendet, wobei 1 Punkt bedeutet, dass das Kriterium nicht erfüllt ist, 2 Punkte bedeuten, dass das Kriterium teilweise erfüllt ist und 3 Punkte bedeuten, dass das Kriterium vollständig erfüllt ist.

Die Punkte werden mit der Gewichtung des jeweiligen Kriteriums multipliziert, um die gewichteten Punkte zu erhalten. Anschliessend werden die gewichteten Punkte aller Kriterien summiert, um die Gesamtpunktzahl für jede Spielidee zu erhalten. Zusätzlich wird überprüft, ob die Spielidee die Ausschlusskriterien erfüllt und damit für die weitere Entwicklung in Betracht gezogen werden kann. Die Bewertung der Spielideen erfolgt durch die Projektgruppe, um eine fundierte Entscheidung über die Auswahl der Spielidee zu treffen.

6.3.1 Strategische Kriterien

Sponsorenintegration

Beschreibung	Das Spiel bietet konkrete, sichtbare und kreative Integrationsmöglichkeiten für Sponsoren.
Skala	• 1 Punkt: Es gibt keine Integrationsmöglichkeiten für Sponsoren • 2 Punkte: Es gibt eine sichtbare Integrationsmöglichkeit für Sponsoren • 3 Punkte: Der Sponsor kann durch gebrandete Spielelemente, sponsorbezogene Aufgaben oder thematisch passende Spielinhalte direkt ins Spielerlebnis eingebunden werden.
Gewichtung	12%

Tabelle 1: Bewertungskriterium: Sponsorenintegration

Spielidee	**Punktzahl**
Stadion-Shootout	3
Stadion-Sprint	3
Stadion-Plays	3
Stadion-Quiz	2
Stadion-Duell	2

Tabelle 2: Bewertung Sponsorenintegration

Kommentar zur Bewertung:
- Bei der Spielidee Stadion-Shootout kann die Bande hinter dem Tor kreativ für den Sponsor gestaltet werden.
- In der Spielidee Stadion-Sprint können die Hindernisse kreativ für den Sponsor gestaltet werden.
- In der Spielidee Stadion-Plays kann das Eisfeld kreativ für den Sponsor gestaltet werden.

Einbindung in die Lakers-App

Beschreibung	Das Spiel kann in die bestehende Lakers-App integriert werden, um eine grössere Reichweite der App zu erreichen.
Skala	• 1 Punkt: Das Spiel kann nicht in die Lakers-App integriert werden • 2 Punkte: Das Spiel kann teilweise in die Lakers-App integriert werden • 3 Punkte: Das Spiel kann vollständig in die Lakers-App integriert werden
Gewichtung	12%

Tabelle 3: Bewertungskriterium: Einbindung in die Lakers-App

Spielidee	**Punktzahl**
Stadion-Shootout	3
Stadion-Sprint	1
Stadion-Plays	3
Stadion-Quiz	3
Stadion-Duell	3

Tabelle 4: Bewertung Einbindung in die Lakers-App

Kommentar zur Bewertung:
- In der Spielidee Stadion-Sprint wird die App nicht verwendet, daher wurde hier eine Punktzahl von 1 vergeben.

Marketingaufwand

Beschreibung	Der organisatorische und kommunikative Aufwand zur Einführung und Bewerbung des Spiels ist gering.
Skala	• 1 Punkt: Der Marketingaufwand ist hoch, da viel Content erstellt werden muss, um das Spiel spannend zu gestalten • 2 Punkte: Der Marketingaufwand ist mittel, da etwas Content erstellt werden muss, um das Spiel spannend zu gestalten • 3 Punkte: Der Marketingaufwand ist gering, da kein Content erstellt werden muss, um das Spiel spannend zu gestalten
Gewichtung	5%

Tabelle 5: Bewertungskriterium: Marketingaufwand

Spielidee	**Punktzahl**
Stadion-Shootout	3
Stadion-Sprint	3
Stadion-Plays	3
Stadion-Quiz	2
Stadion-Duell	3

Tabelle 6: Bewertung Marketingaufwand

Kommentar zur Bewertung:

- Bei der Spielidee Stadion-Quiz muss Content erstellt werden, um die Fragen spannend zu gestalten, dieser Aufwand wird sich auf eine oder zwei Stunden vor jedem Spiel belaufen. Daher wurde hier eine Punktzahl von 2 vergeben.

6.3.2 Organisatorische Kriterien

Anzahl der Teilnehmenden

Beschreibung	Das Spiel ist flexibel mit unterschiedlicher Anzahl der Teilnehmenden (Einzelperson bis Stadionbeteiligung) durchführbar.
Skala	• 1 Punkt: Die Anzahl der Teilnehmenden ist nicht skalierbar • 2 Punkte: Die Anzahl der Teilnehmenden ist bis 100 skalierbar • 3 Punkte: Die Anzahl der Teilnehmenden ist auf die Stadiongrösse skalierbar
Gewichtung	10%

Tabelle 7: Bewertungskriterium: Anzahl der Teilnehmenden

Spielidee	**Punktzahl**
Stadion-Shootout	3
Stadion-Sprint	2
Stadion-Plays	3
Stadion-Quiz	3
Stadion-Duell	2

Tabelle 8: Bewertung Anzahl der Teilnehmenden

Kommentar zur Bewertung:
- Für die Spielidee Stadion-Sprint müssen viele Personen gleichzeitig spielen, damit die Lautstärke genügend signifikant ist, daher wurde hier eine Punktzahl von 2 vergeben.
- Für die Spielidee Stadion-Duell können maximal 256 Personen mitspielen, damit es mit der Spielzeit aufgeht, daher wurde hier eine Punktzahl von 2 vergeben.

Zielgruppeninklusion

Beschreibung	Das Spiel ist ohne alters-, sprach- oder geschlechterspezifische Barrieren verständlich und spielbar
Skala	• 1 Punkt: Das Spiel ist nur für eine bestimmte Zielgruppe verständlich und spielbar • 2 Punkte: Das Spiel ist für mehrere Zielgruppen verständlich und spielbar, aber es gibt noch Barrieren für bestimmte Zielgruppen • 3 Punkte: Das Spiel ist für alle Zielgruppen verständlich und spielbar, ohne Barrieren
Gewichtung	17%

Tabelle 9: Bewertungskriterium: Zielgruppeninklusion

Spielidee	Punktzahl
Stadion-Shootout	3
Stadion-Sprint	2
Stadion-Plays	2
Stadion-Quiz	3
Stadion-Duell	2

Tabelle 10: Bewertung Zielgruppeninklusion

Kommentar zur Bewertung:

- Bei der Spielidee Stadion-Sprint werden Personen mit leisen Stimmen benachteiligt, da die Lautstärke dafür ausschlaggebend ist, daher wurde hier eine Punktzahl von 2 vergeben.
- Bei den Spielideen Stadion-Plays und Stadion-Duell wird Vorwissen von Computerspielen vorausgesetzt, daher wurde hier eine Punktzahl von 2 vergeben.

Verständlichkeit des Spielkonzepts

Beschreibung	Das Spielprinzip ist intuitiv verständlich und erfordert keine ausführliche Erklärung.
Skala	• 1 Punkt: Das Spielprinzip ist nicht intuitiv verständlich und erfordert eine ausführliche Erklärung • 2 Punkte: Das Spielprinzip ist teilweise intuitiv verständlich und erfordert eine kurze Erklärung • 3 Punkte: Das Spielprinzip ist intuitiv verständlich und erfordert keine ausführliche Erklärung
Gewichtung	17%

Tabelle 11: Bewertungskriterium: Verständlichkeit des Spielkonzepts

Spielidee	Punktzahl
Stadion-Shootout	2
Stadion-Sprint	2
Stadion-Plays	1
Stadion-Quiz	3
Stadion-Duell	2

Tabelle 12: Bewertung Verständlichkeit des Spielkonzepts

Kommentar zur Bewertung:

- Bei der Spielidee Stadion-Shootout wurde eine Punktzahl von 2 vergeben. Zwar ist das Grundprinzip eines Schussspiels grundsätzlich bekannt, jedoch ist nicht unmittelbar ersichtlich, wie die Zuschauenden den Schuss beeinflussen, welche Eingabe zu welchem Effekt führt und wie daraus eine gewinnende Person bestimmt wird. Da diese Regeln vor Spielbeginn zusätzlich erklärt werden müssten, wurde die Verständlichkeit als eingeschränkt bewertet.
- Bei der Spielidee Stadion-Sprint wurde eine Punktzahl von 2 vergeben. Das Ziel des Spiels, eine Figur möglichst schnell vorwärtszubewegen, ist einfach verständlich. Unklar bleibt jedoch, wie genau die Steuerung über Lautstärke funktioniert und welchen Einfluss einzelne Zuschauende auf das Gesamtergebnis haben. Zudem ist fraglich, ob einzelne Stimmen im Stadion einen messbaren Unterschied erzeugen können. Dadurch besteht das Risiko, dass Teilnehmende den Zusammenhang zwischen Aktion und Spielreaktion nicht eindeutig erkennen.
- Bei der Spielidee Stadion-Plays wurde eine Punktzahl von 1 vergeben. Die Spielidee setzt voraus, dass Zuschauende verstehen, welche Eingaben sie machen können und wie diese Eingaben das Spielgeschehen beeinflussen. Dieser Zusammenhang ist bei der Idee nicht unmittelbar erkennbar. Ohne zusätzliche Erklärung bleibt unklar, ob die Teilnehmenden einzelne Aktionen auswählen, gemeinsam abstimmen oder direkt eine Spielfigur steuern.
- Bei der Spielidee Stadion-Duell wurde eine Punktzahl von 2 vergeben. Das Duellprinzip ist grundsätzlich verständlich, jedoch ist nicht klar, wie die Zuschauenden einer Seite zugeordnet werden und wie ihre Eingaben den Ausgang des Duells beeinflussen. Dadurch müsste vor Spielbeginn erklärt werden, welche Rolle die Teilnehmenden übernehmen und wie ihr Verhalten zum Spielergebnis beiträgt.

Scoreboard-Integration

Beschreibung	Die Spielidee bietet die Möglichkeit, ein Scoreboard zu integrieren, damit die gewinnende Person des Spiels einen Preis erhalten kann.
Skala	• 1 Punkt: Es gibt keine Möglichkeit, ein Scoreboard zu integrieren und eine gewinnende Person zu bestimmen • 2 Punkte: Es gibt eine Möglichkeit, ein Scoreboard zu integrieren, aber es ist nicht möglich, eine gewinnende Person zu bestimmen • 3 Punkte: Es gibt eine Möglichkeit, ein Scoreboard zu integrieren, und es ist möglich, eine gewinnende Person zu bestimmen
Gewichtung	5%

Tabelle 13: Bewertungskriterium: Scoreboard-Integration

Spielidee	Punktzahl
Stadion-Shootout	1
Stadion-Sprint	1
Stadion-Plays	1
Stadion-Quiz	3
Stadion-Duell	3

Tabelle 14: Bewertung Scoreboard-Integration

Kommentar zur Bewertung:

- Beim Stadion-Sprint basiert der Spielfortschritt auf kollektiver Lautstärke. Da mehrere Personen gleichzeitig beitragen und einzelne Beiträge nicht getrennt messbar sind, eignet sich das Konzept kaum für ein individuelles Scoreboard.
- Bei Stadion-Plays und Stadion-Shootout steht die gemeinsame Auswahl oder Steuerung von Spielzügen im Vordergrund. Da die Wirkung einzelner Teilnehmender auf das Ergebnis nicht eindeutig bestimmbar ist, lässt sich kein nachvollziehbare gewinnende Person ableiten.

6.3.3 Technische Kriterien

Technische Machbarkeit

Beschreibung	Das Spiel ist mit bestehender Infrastruktur der Lakers Sport AG realisierbar.
Skala	• 1 Punkt: Das Spiel ist mit der bestehenden Infrastruktur der Lakers Sport AG nicht realisierbar und neue Hardware oder Software muss angeschafft werden • 2 Punkte: Das Spiel ist mit der bestehenden Infrastruktur der Lakers Sport AG teilweise realisierbar, es müssen jedoch Anpassungen an der Infrastruktur vorgenommen werden • 3 Punkte: Das Spiel ist mit der bestehenden Infrastruktur der Lakers Sport AG vollständig realisierbar, es müssen keine Anpassungen an der Infrastruktur vorgenommen werden
Gewichtung	12%

Tabelle 15: Bewertungskriterium: Technische Machbarkeit

Spielidee	**Punktzahl**
Stadion-Shootout	3
Stadion-Sprint	1
Stadion-Plays	3
Stadion-Quiz	3
Stadion-Duell	3

Tabelle 16: Bewertung Technische Machbarkeit

Kommentar zur Bewertung:

- Bei der Spielidee Stadion-Sprint ist es nicht möglich, die bestehende Infrastruktur der Lakers Sport AG zu verwenden, da die Spielfigur mit einem Mikrofon gesteuert wird, daher wurde hier eine Punktzahl von 1 vergeben.

Entwicklungsaufwand

Beschreibung	Der Entwicklungsaufwand für einen funktionsfähigen Prototyp liegt im zeitlichen Rahmen der Bachelorarbeit.
Skala	• 1 Punkt: Der Entwicklungsaufwand für einen funktionsfähigen Prototyp liegt deutlich über dem zeitlichen Rahmen der Bachelorarbeit • 2 Punkte: Der Entwicklungsaufwand für einen funktionsfähigen Prototyp liegt leicht über dem zeitlichen Rahmen der Bachelorarbeit, es ist jedoch möglich den Prototyp innerhalb des zeitlichen Rahmens zu entwickeln • 3 Punkte: Der Entwicklungsaufwand für einen funktionsfähigen Prototyp liegt im zeitlichen Rahmen der Bachelorarbeit, es ist realistisch den Prototyp innerhalb des zeitlichen Rahmens zu entwickeln
Gewichtung	10%

Tabelle 17: Bewertungskriterium: Entwicklungsaufwand

Spielidee	**Punktzahl**
Stadion-Shootout	3
Stadion-Sprint	3
Stadion-Plays	1
Stadion-Quiz	3
Stadion-Duell	1

Tabelle 18: Bewertung Entwicklungsaufwand

Kommentar zur Bewertung:

- Bei den Spielideen Stadion-Plays und Stadion-Duell ist es nicht realistisch, einen funktionsfähigen Prototyp innerhalb des zeitlichen Rahmens der Bachelorarbeit zu entwickeln, da diese Spielideen sehr komplex sind. Daher wurde hier eine Punktzahl von 1 vergeben.

6.3.4 Ausschlusskriterium

Beinhaltet keine No-Gos

Beschreibung	Die Spielidee beinhaltet keine No-Gos, welche von der Lakers Sport AG definiert wurden. No-Gos sind Kriterien, welche ein Spiel nicht erfüllen darf, damit es für die Lakers infrage kommt. Diese No-Gos sind: • Politische Themen • Religiöse Themen • Gewalt • Glücksspiel • Themen, die mit dem Gesetz in Konflikt stehen
Skala	Ausschlusskriterium: Ein Spiel, welches eines der No-Gos beinhaltet, wird von der Evaluation ausgeschlossen und erhält eine Totalpunkteanzahl von 0 Punkten

Tabelle 19: Bewertungskriterium: Beinhaltet keine No-Gos

Spielidee	**Bewertung**
Stadion-Shootout	Die Spielidee enthält keine No-Gos.
Stadion-Sprint	Die Spielidee enthält keine No-Gos.
Stadion-Plays	Die Spielidee enthält keine No-Gos.
Stadion-Quiz	Die Spielidee enthält keine No-Gos.
Stadion-Duell	Die Spielidee enthält keine No-Gos.

Tabelle 20: Bewertung: Beinhaltet keine No-Gos

Spieldauer

Beschreibung	Die Spielidee kann inklusive Erklärung und Auswertung innerhalb von maximal vier Minuten durchgeführt werden.
Skala	Ausschlusskriterium: Ein Spiel, welches über eine Spieldauer von mehr als vier Minuten verfügt, wird von der Evaluation ausgeschlossen und erhält eine Totalpunkteanzahl von 0 Punkten

Tabelle 21: Bewertungskriterium: Spieldauer

Spielidee	Bewertung
Stadion-Shootout	Die Spielidee kann nicht in der vorgegebenen Zeit in der Pause durchgeführt werden.
Stadion-Sprint	Die Spielidee kann in der vorgegebenen Zeit in der Pause durchgeführt werden.
Stadion-Plays	Die Spielidee kann nicht in der vorgegebenen Zeit in der Pause durchgeführt werden.
Stadion-Quiz	Die Spielidee kann in der vorgegebenen Zeit in der Pause durchgeführt werden.
Stadion-Duell	Die Spielidee kann nicht in der vorgegebenen Zeit in der Pause durchgeführt werden.

Tabelle 22: Bewertung Spieldauer

6.3.5 Totalbewertung der Spielideen

Die Ideen, welche eine Gesamtpunktzahl von 0 Punkten erhalten haben, wurden von der Evaluation ausgeschlossen, da sie eines der Ausschlusskriterien nicht erfüllt haben.

Spielidee	Total Punkte strategische Kriterien	Total Punkte organisatorische Kriterien	Total Punkte technische Kriterien	Punkte Total
Stadion-Shootout	**87**	120	**66**	0
Stadion-Sprint	63	93	42	198
Stadion-Plays	**87**	86	46	0
Stadion-Quiz	70	**147**	**66**	**283**
Stadion-Duell	75	103	46	0

Tabelle 23: Gesamtbewertung der Spielideen

6.3.6 Auswahl der Spielidee

Durch die Bewertung der Spielideen anhand der definierten Bewertungskriterien erreichte die Spielidee „Stadion-Quiz“ die höchste Gesamtpunktzahl. Dabei wurden die gewichteten Punkte aller Kriterien zusammengerechnet. Die Idee Stadion-Quiz erreichte 283 Punkte. Der Abstand zu den anderen Ideen ergibt sich insbesondere aus den organisatorischen Kriterien.

6.4 Mitmachwert

Um die Publikumszahlen während der zweiten Pause zu analysieren, wurden zwei Spiele besucht und die Tribünen fotografisch dokumentiert. Die Fotos wurden anschliessend ausgewertet und die Anzahl der anwesenden Personen gezählt. Die ausgewerteten Aufnahmen sind im Anhang unter Kapitel A.5 dokumentiert. Aus dieser Zählung ergibt sich, dass während der zweiten Pause im vorgesehenen Zeitslot durchschnittlich 716 Personen in den Sektoren B und C befinden. Bei einer maximalen Stadionkapazität von 6100 Plätzen entspricht dies rund 11,7 % der gesamten verfügbaren Plätze im Stadion.

Die folgende Tabelle fasst die gezählten Personen pro Aufnahme zusammen.

Spieltag	**Zeitpunkt ab Pausebeginn**	**Sektor B**	**Sektor C**	**Total B + C**
27.02.2026	1 Minute	290	500	790
27.02.2026	3 Minuten	231	354	585
05.03.2026	1 Minute	307	630	937
05.03.2026	3 Minuten	190	460	650
05.03.2026	4 Minuten	174	446	620
Durchschnitt		238	478	**716**

Tabelle 24: Auswertung der Publikumszahlen in den Sektoren B und C während der zweiten Pause

Für die Bewertung der späteren Nutzung wird der Mitmachwert als Prozentanteil der aktiven Teilnehmenden an den anwesenden Personen in den betrachteten Sektoren während des relevanten Zeitfensters definiert. Der Wert wird in Prozent nach folgender Formel berechnet:

$$\text{Mitmachwert} = \frac{\text{aktive Teilnehmende}}{\text{anwesende Personen in den betrachteten Sektoren}} \times 100$$

Als Bezugsgrösse wird nicht die maximale Stadionkapazität von 6100 Plätzen verwendet, sondern die durchschnittlich anwesende Personenzahl in den betrachteten Sektoren während des relevanten Zeitfensters. Da das Spiel für alle Altersgruppen geeignet ist, wird diese Personenzahl vollständig als erreichbare Zielgruppe betrachtet. Damit bezieht sich der Mitmachwert auf die tatsächlich erreichbare Zielgruppe und nicht auf Personen, die sich zu diesem Zeitpunkt nicht im beobachteten Bereich befinden.

Die Schwellenwerte werden als projektbezogene Operationalisierung festgelegt und sind nicht als allgemeingültiger Benchmark zu verstehen. Im Kickoff wurde festgehalten, dass bereits eine Beteiligung von etwa 50 Personen in Ordnung wäre. Bei 716 anwesenden Personen entspricht dies rund 7 %. Da die Teilnahme freiwillig ist und die Aufmerksamkeit der Stadionbesuchenden während der Pause auf verschiedene Aktivitäten verteilt ist, kann keine vollständige Beteiligung erwartet werden. Gleichzeitig soll ein guter Mitmachwert deutlich über einer vereinzelten Nutzung liegen. Für diese Arbeit wird ein guter Mitmachwert deshalb ab 15 % festgelegt. Bei einer durchschnittlichen Anwesenheit von 716 Personen entspricht dies rund 107 aktiven Teilnehmenden.

Bewertung	Mitmachwert	Interpretation
Sehr niedrig	unter 5 %	Das Angebot wird nur vereinzelt genutzt.
Niedrig	5 % bis 9 %	Es besteht eine erkennbare, aber geringe Nutzung.
Mittel	10 % bis 14 %	Das Angebot erreicht einen wahrnehmbaren Teil der Zielgruppe.
Gut	15 % bis 24 %	Das Angebot wird von einem relevanten Anteil der Zielgruppe genutzt.
Sehr gut	ab 25 %	Das Angebot erreicht eine hohe Beteiligung innerhalb des betrachteten Stadionbereichs.

Tabelle 25: Bewertungsskala für den Mitmachwert des Pausenspiels

7 Konzept

Im folgenden Kapitel wird das Konzept beschrieben. In den ersten Unterkapiteln wird auf die Anwendungsfälle und nichtfunktionalen Anforderungen eingegangen. In den folgenden drei Unterkapiteln wird auf die drei Spielmodi Quiz, Kreuzworträtsel und Bilderrätsel eingegangen. Darauf folgt die mobile Teilnehmendenansicht, welche die teilnehmende Person auf ihrem mobilen Gerät sieht. Im Kapitel zum Administrationsbereich wird erläutert, wie Spiele verwaltet werden. Im letzten Unterkapitel werden die Mockups als visuelle Entwürfe genauer dargestellt.

7.1 Anwendungsfälle

In diesem Kapitel werden die Anwendungsfälle für das Projekt beschrieben. Die Anwendungsfälle beschreiben die unterschiedlichen Ziele verschiedener Akteure, die mit dem interaktiven Pausenspielsystem interagieren. Sie geben einen Überblick über die Funktionen und Interaktionen, die das System bereitstellen muss, um die Bedürfnisse der verschiedenen Akteure zu erfüllen.

7.1.1 Akteure

Für das interaktives Pausenspielsystem gibt es drei verschiedene Akteure.

Zuschauende
Zuschauende sind Personen, welche sich im Stadion befinden und das interaktive Pausenspielsystem verwenden.

Marketing
Das Marketing ist eine Person, welche die Inhalte für das Interaktive Pausenspielsystem verwaltet.

Spielleitung
Die Spielleitung bedient das interaktive Pausenspielsystem.

Aus diesen Rollen ergibt sich die Aufteilung der Oberfläche in drei Bereiche: den Administrationsbereich, die Präsentationsansicht für den Videowürfel und die mobile Teilnehmendenansicht. Diese Trennung folgt dem Spielablauf. Das Marketing bereitet Inhalte vor, die Spielleitung steuert Durchläufe im Administrationsbereich, die Zuschauenden sehen die Inhalte auf dem Videowürfel und geben Antworten über das eigene Mobilgerät ab. Damit werden Vorbereitung, gemeinsame Orientierung im Stadion und individuelle Teilnahme bewusst voneinander getrennt. Die daraus abgeleiteten Anwendungsfälle sind im Diagramm zusammengefasst (siehe Abbildung 14).

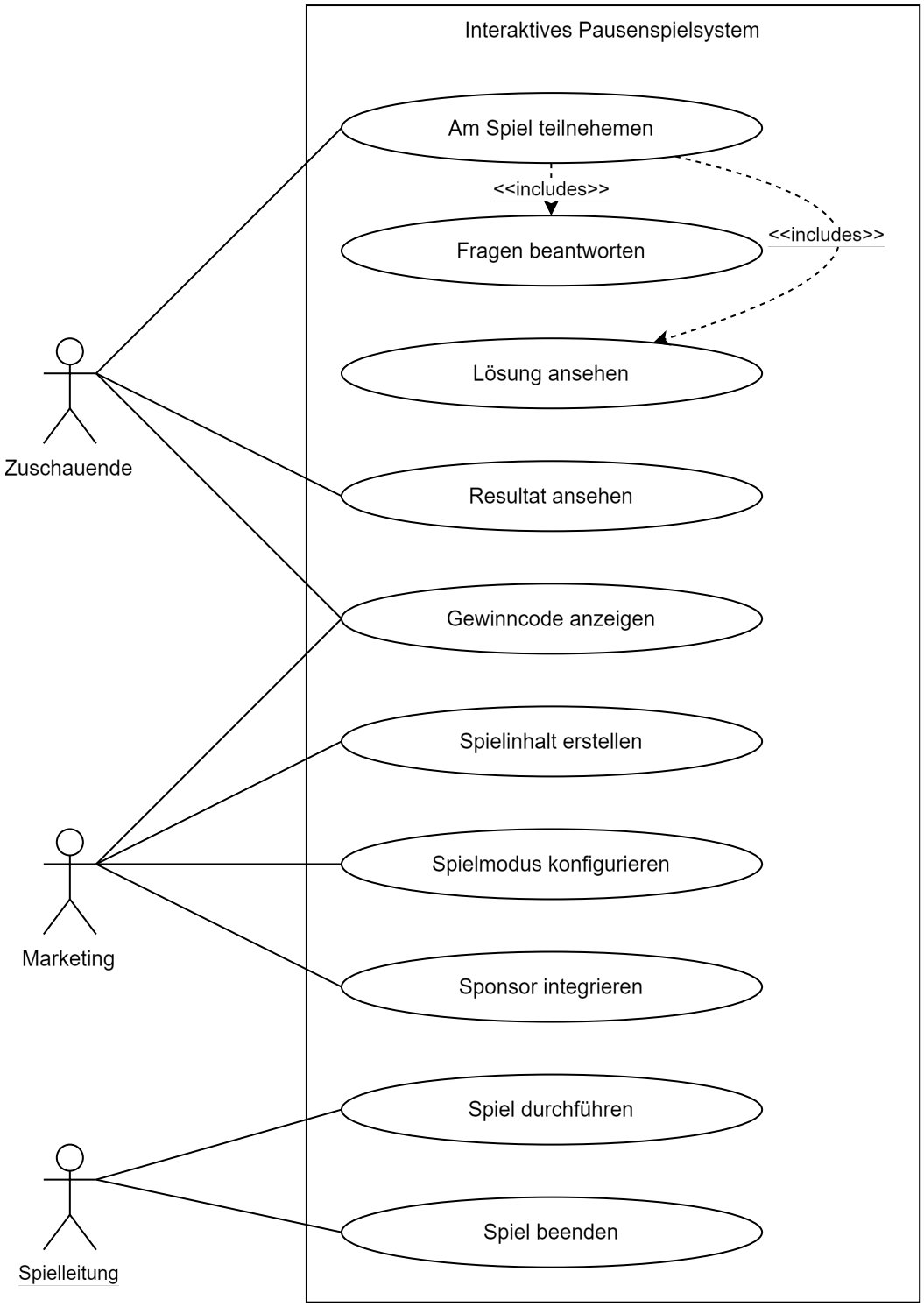


Abbildung 14: Anwendungsfall Diagram
Quelle: Eigene Darstellung

Aus den Anwendungsfällen ergeben sich die funktionalen Anforderungen, welche jeweils in den entsprechenden Anwendungsfällen beschrieben werden. Die Anwendungsfälle werden in folgende Gruppen unterteilt:

- Zuschauende: UC0X
- Marketing: UC1X
- Spielleitung: UC2X

7.1.2 UC01 - Am Spiel teilnehemen

Beschreibung

Die Zuschauenden können am Spiel teilnehmen, indem sie in der SCRJ Lakers App auf die entsprechende Funktion klicken.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR011	App Teilnahme	In der SCRJ Lakers App können Zuschauende am Spiel teilnehmen.	Nein

Tabelle 26: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC01 abgeleitet wurden

7.1.3 UC02 - Fragen beantworten

Beschreibung

Die Zuschauenden können in der Applikation die Fragen des Spiels beantworten, indem sie auf die entsprechenden Antwortmöglichkeiten klicken.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR021	Fragen beantworten	Die Zuschauenden können in der Applikation die Fragen des Spiels beantworten.	Ja

Tabelle 27: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC02 abgeleitet wurden

7.1.4 UC03 - Lösung ansehen

Beschreibung

Die Zuschauenden können auf dem Videowürfel die Lösungen der Fragen sehen.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR031	Lösung ansehen	Die Zuschauenden können auf dem Videowürfel die Lösungen der Fragen sehen.	Ja

Tabelle 28: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC03 abgeleitet wurden

7.1.5 UC04 - Resultat anzeigen

Beschreibung

Die Zuschauenden können ein Scoreboard sehen, auf dem sie sich mit anderen vergleichen können.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR041	Spiel Scoreboard ansehen	Die Zuschauenden können ein Spiel Scoreboard sehen, auf dem sie sich mit anderen vergleichen können.	Ja
FR042	Saison Scoreboard ansehen	Die Zuschauenden können ein Saison Scoreboard sehen, auf dem sie sich mit anderen vergleichen können.	Nein

Tabelle 29: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC04 abgeleitet wurden

7.1.6 UC05 - Gewinncode anzeigen

Beschreibung

Die Zuschauenden können sehen, ob sie einen Gewinn erzielt haben oder nicht.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR051	Gewinn anzeigen	Die Zuschauenden können sehen, ob sie einen Gewinn erzielt haben.	Ja

Tabelle 30: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC05 abgeleitet wurden

7.1.7 UC11 - Spielinhalte erstellen

Beschreibung

Das Marketing kann neue Spielinhalte erstellen, neue Fragen und Antworten in das System einfügen.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR111	Quizfragen erfassen	Es können neue Quizfragen durch das Marketing erstellt werden.	Ja
FR112	Quizantworten erfassen	Es können neue Quizantworten erstellt und die richtige durch das Marketing definiert werden.	Ja
FR013	Kreuzworträtsel erfassen	Es können neue Kreuzworträtsel durch das Marketing erstellt werden.	Ja
FR014	Bilder erfassen	Es können neue Bilder durch das Marketing erfasst werden.	Ja
FR015	Bildeffekt anpassen	Der Bildeffekt kann ausgewählt werden.	Nein

Tabelle 31: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC11 abgeleitet wurden

7.1.8 UC12 - Spielmodus konfigurieren

Beschreibung

Das Marketing kann den Spielmodus konfigurieren und auf die Bedürfnisse anpassen.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR121	Spielmoduskonfiguration	Es kann zwischen verschiedenen Spielmodi gewählt werden.	Ja
FR122	Spieldauer anpassen	Die Spiel- sowie Rundendauer kann angepasst werden.	Nein

Tabelle 32: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC12 abgeleitet wurden

7.1.9 UC13 - Sponsor integrtieren

Beschreibung

Das Marketing kann einen Sponsor integrieren, damit dieser während des Spiels sichtbar ist.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR131	Sponsor integrieren	Ein neuer Sponsor kann in das Spiel integriert werden.	Ja

Tabelle 33: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC13 abgeleitet wurden

7.1.10 UC21 - Spiel durchführen

Beschreibung

Die Spielleitung kann das Spiel durchführen, indem sie die verschiedenen Funktionen des Systems bedient.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR211	Spiel laden	Die Spielleitung kann das Spiel laden.	Ja
FR212	Spiel starten	Die Spielleitung kann das Spiel starten.	Ja
FR213	Nächste Frage anzeigen	Die Spielleitung kann die nächste Frage anzeigen.	Nein
FR214	Spiel pausieren	Die Spielleitung kann das Spiel pausieren.	Nein

Tabelle 34: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC21 abgeleitet wurden

7.1.11 UC22 - Spiel beenden

Beschreibung

Die Spielleitung kann das Spiel beenden.

Abgeleitete funktionale Anforderungen

ID	Name	Beschreibung	Zwingend
FR221	Spiel beenden	Die Spielleitung kann das Spiel beenden.	Ja
FR222	Spiel auswerten	Die Spielleitung kann das Spiel auswerten.	Nein

Tabelle 35: Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC22 abgeleitet wurden

7.2 Nichtfunktionale Anforderungen

In diesem Kapitel sind die nichtfunktionalen Anforderungen nach FURPS dokumentiert. (Musa & Alkhateeb, 2013)

7.2.1 Erklärung

FURPS unterteilt die Qualitätsanforderungen an Software in folgende Gruppen:

- Funktionalität - NFR0XX
- Bedienbarkeit - NFR1XX
- Zuverlässigkeit - NFR2XX
- Leistung - NFR3XX
- Wartbarkeit - NFR4XX

Je tiefer die Zahl bei der Priorisierung ist, desto wichtiger ist die Anforderung.

7.2.2 Funktionalität

ID	Name	Akzeptanzkriterium	Verifizierung	Priorität
NFR001	Der Zugriff auf den Administrationsbereich muss auf autorisierte Personengruppen beschränkt sein.	Personen ohne Administrationsberechtigung können den Administrationsbereich nicht aufrufen.	Zugriffstest mit berechtigten und nicht berechtigten Konten	1
NFR002	Nur gestartete Spiele dürfen für Zuschauende über den QR-Code erreichbar sein.	Nicht gestartete Spiele können von Zuschauenden nicht gespielt werden.	Funktionstest mit gestarteten und nicht gestarteten Spielen	1
NFR003	Spielinhalte im Administrationsbereich sollen als Entwurf gespeichert werden können, auch wenn sie noch nicht vollständig sind.	Ein Entwurf kann trotz unvollständiger Inhalte gespeichert werden. Vor dem Starten oder Veröffentlichen eines Spiels werden jedoch fehlende Pflichtangaben erkannt und ausgewiesen.	Funktionstest mit unvollständigen Entwürfen sowie mit vollständigen und unvollständigen spielbereiten Inhalten	2
NFR004	Innerhalb des Administrationsbereichs müssen Funktionen entsprechend der jeweiligen Rolle freigegeben sein.	Nur die mit der Administrationsrolle ausgestatteten Personen können auf die entsprechenden Funktionen zugreifen.	Zugriffstest auf Funktionsbereiche mit den Rollen Marketing und Spielleitung	2

Tabelle 36: Funktionale Aspekte der nichtfunktionalen Anforderungen

7.2.3 Bedienbarkeit

ID	Name	Akzeptanzkriterium	Verifizierung	Priorität
NFR101	Der Einstieg in das Spiel muss für Stadionbesuchende mit wenigen, selbsterklärenden Schritten möglich sein.	Mindestens 3 von 4 Testpersonen können nach dem Scannen des QR-Codes ohne mündliche Erklärung innerhalb von höchstens 30 Sekunden einem aktiven Spiel beitreten und eine Antwort abgeben.	Usability-Test mit Testpersonen aus der Zielgruppe und Messung von Erfolgsquote und Bearbeitungszeit	1
NFR102	Die Spieloberfläche auf dem Smartphone muss auch unter Stadionbedingungen klar lesbar und bedienbar sein.	Texte, Antwortoptionen und zentrale Bedienelemente sind auf einem üblichen Smartphone ohne Zoomen gut lesbar und per Touch bedienbar.	Visuelle Prüfung auf verschiedenen Smartphone-Displays sowie Bedienungstest unter typischen Nutzungsbedingungen	1

Tabelle 37: Benutzbarkeitsaspekte der nichtfunktionalen Anforderungen

7.2.4 Zuverlässigkeit

ID	Name	Akzeptanzkriterium	Verifizierung	Priorität
NFR201	Ein aktives Spiel muss bei einzelnen Fehlbedienungen oder Verbindungsabbrüchen von Zuschauenden weiterlaufen.	Wenn einzelne Zuschauende die Verbindung verlieren oder eine ungültige Anfrage senden, bleibt das Spiel für die übrigen Teilnehmenden und die Spielleitung weiterhin nutzbar.	Integrationstest mit simulierten Verbindungsabbrüchen und ungültigen Anfragen einzelner Teilnehmender	1
NFR202	Spielstände und Resultate eines laufenden Spiels dürfen während der Durchführung nicht verloren gehen.	Nach einem Seiten-Reload im Administrationsbereich oder einer kurzen Unterbrechung bleiben der aktuelle Spielstatus und bereits erfasste Antworten erhalten oder können wiederhergestellt werden.	Integrationstest mit Seiten-Reload und kurzer Unterbrechung während eines laufenden Spiels	2

Tabelle 38: Zuverlässigkeitsaspekte der nichtfunktionalen Anforderungen

7.2.5 Leistung

ID	Name	Akzeptanzkriterium	Verifizierung	Priorität
NFR301	Zuschauende müssen nach dem Scannen des QR-Codes innerhalb kurzer Zeit dem aktiven Spiel beitreten können.	Die Startseite des aktiven Spiels ist unter normalen Bedingungen innerhalb von höchstens 3 Sekunden geladen.	Messung der Ladezeit auf mobilen Endgeräten unter repräsentativen Bedingungen	1
NFR302	Antworten der Zuschauenden müssen so schnell verarbeitet werden, dass das Spiel im Zeitfenster (4 Minuten) der Drittelpause flüssig durchgeführt werden kann.	Eine abgegebene Antwort wird innerhalb von höchstens 2 Sekunden im System erfasst und für die Auswertung berücksichtigt.	Performancetest während einer simulierten Spielrunde	1
NFR303	Das System muss für viele gleichzeitige Teilnehmende im Stadion ausgelegt sein.	Bei einer Lastsimulation mit mindestens 500 gleichzeitigen Teilnehmenden bleiben Beitritt und Antwortabgabe ohne kritische Fehler möglich.	Lasttest mit simulierten gleichzeitigen Zugriffen und Auswertung der Fehlerrate	2

Tabelle 39: Leistungsaspekte der nichtfunktionalen Anforderungen

7.2.6 Wartbarkeit

ID	Name	Akzeptanzkriterium	Verifizierung	Priorität
NFR401	Neue Spielinhalte und weitere Spielmodi sollen mit möglichst geringem Anpassungsaufwand ergänzt werden können.	Neue Quizinhalte oder ein zusätzlicher Spielmodus können über klar definierte Datenstrukturen oder Schnittstellen ergänzt werden, ohne bestehende Kernfunktionen grundlegend umzubauen.	Code-Review der Architektur sowie Prüfung der Datenstrukturen und Schnittstellenbeschreibung	3
NFR402	Konfigurationen für Spiele, Sponsoren und Laufzeiten sollen zentral pflegbar sein.	Anpassungen an Spielinhalten, Sponsorintegration oder Rundendauer können ohne Codeänderung über den Administrationsbereich oder eine zentrale Konfiguration vorgenommen werden.	Funktionstest durch Konfigurationsänderungen ohne Codeanpassung sowie Review der Konfigurationsmöglichkeiten	2
NFR403	Die Identifikation von Zuschauenden soll erweiterbar sein, sodass neben einem Sessiontoken auch eine Anbindung an die Lakers-App über eine definierte API möglich ist.	Das System kann Benutzende sowohl über den bestehenden Sessiontoken als auch über eine bereitgestellte API der Lakers-App eindeutig identifizieren, ohne dass die Kernlogik des Spiels angepasst werden muss.	Code-Review der Schnittstelle sowie Integrationstest der Identifikation über beide Varianten	3

Tabelle 40: Wartbarkeitsaspekte der nichtfunktionalen Anforderungen

7.3 Gemeinsame Spielauswertung

Alle Spielmodi verwenden dieselbe grundlegende Auswertungslogik. Die gewinnende Person wird am Ende des Spiels anhand der Anzahl korrekter Antworten ermittelt. Falls mehrere Teilnehmende dieselbe Punktzahl erreichen, wird die gewinnende Person aus dieser Gruppe ausgelost. Die gewinnende Person erhält in der mobilen Teilnehmendenansicht ein Formular und erfasst dort Name und E-Mail-Adresse, damit der Gewinn eindeutig zugeordnet und im Fanshop abgeholt werden kann.

7.3.1 Spielmodus Quiz

Beim Spielmodus „Quiz“ werden jeweils fünf Quizfragen gestellt. Zu jeder Frage gibt es vier Antwortmöglichkeiten, von denen nur eine korrekt ist. Die Zuschauenden haben pro Frage 20 Sekunden Zeit, um die richtige Antwort auszuwählen. Danach wird die richtige Antwort angezeigt und die Punkte entsprechend vergeben. Die Ermittlung der gewinnenden Person folgt der gemeinsamen Spielauswertung aus Kapitel 7.3.

7.3.2 Spielmodus Kreuzworträtsel

Beim Spielmodus „Kreuzworträtsel“ werden jeweils fünf Fragen gestellt, diese werden auf dem Videowürfel angezeigt. Zuschauende sehen in der mobilen Teilnehmendenansicht ein Kreuzworträtsel, das aus fünf Feldern besteht, die den Fragen entsprechen. Pro Frage gibt es eine Antwort, die in das entsprechende Feld des Kreuzworträtsels eingetragen werden muss. Nach Ablauf der Zeit werden die Antworten auf dem Videowürfel angezeigt und für die korrekten Antworten Punkte verteilt. Die Ermittlung der gewinnenden Person folgt der gemeinsamen Spielauswertung aus Kapitel 7.3.

7.3.3 Spielmodus Bilderrätsel

Beim Spielmodus „Bilderrätsel“ werden jeweils drei Bilder gezeigt. Dabei soll es vorgegebene Filter (z.B. verpixelt, verdreht oder invertiert) geben, welche die Erkennung der Bilder erschweren. Zu jedem Bild gibt es vier Antwortmöglichkeiten, von denen nur eine korrekt ist. Die Bilder werden auf dem Videowürfel angezeigt, während die Antwortmöglichkeiten auf dem Videowürfel und in der mobilen Teilnehmendenansicht zu sehen sind. Die Zuschauenden haben pro Bild 20 Sekunden Zeit, um die richtige Antwort auszuwählen. Danach wird die richtige Antwort auf dem Videowürfel angezeigt und die Punkte entsprechend vergeben. Die Ermittlung der gewinnenden Person folgt der gemeinsamen Spielauswertung aus Kapitel 7.3.

7.4 Mobile Teilnehmendenansicht

Zur Abklärung der Integrationsmöglichkeiten in die SCRJ Laker-App wurde ein Gespräch mit der Firma soul.media, der Entwicklerin der Lakers-App, geführt. Dabei wurden zwei Möglichkeiten für die Weiterentwicklung des Projekts identifiziert:

1. Die Firma soul.media übernimmt die Entwicklung des Backends und die Integration in die Lakers-App, während sich die Bachelorarbeit auf die Entwicklung der Views für den Videowürfel konzentriert.
2. Das Backend und die mobile Teilnehmendenansicht werden im Rahmen der Bachelorarbeit entwickelt und die Integration in die Lakers-App wird als zukünftige Erweiterung betrachtet.

Nach der Abwägung der Vor- und Nachteile wurde gemeinsam mit dem Betreuer entschieden, die zweite Option zu verfolgen. Diese Variante bietet mehr Kontrolle über die Entwicklung und ermöglicht es, das Projekt künftig gezielt weiterzuentwickeln und zu erweitern.

7.5 Administrationsbereich

Der Administrationsbereich ist ein Bereich, der den Administrierenden ermöglicht, die erstellten Spiele zu verwalten und neue Spiele zu erstellen. Er ist über eine separate URL zugänglich. Als Hauptansicht wird eine Liste mit allen erstellten Spielen angezeigt.

Verwaltung von Spielen

Aus der Liste heraus gibt es die Möglichkeit, ein neues Spiel zu erstellen. Dazu muss die administrierende Person die erforderlichen Informationen eingeben, wie z.B. den Namen des Spiels, ein Datum und den Spielmodus (Quiz, Kreuzworträtsel oder Bilderrätsel). Je nach ausgewähltem Spielmodus müssen weitere Informationen eingegeben werden, wie z.B. die Fragen und Antworten für den Quizmodus, die Bilder und Antworten für den Bilderrätselmodus oder die Fragen und Antworten für den Kreuzworträtselmodus. Nach der Eingabe aller erforderlichen Informationen kann das Spiel gespeichert und in der Liste der Spiele angezeigt werden. Ebenso kann in jedem Spiel ein Sponsor ausgewählt werden, damit dieser auf dem Videowürfel angezeigt wird, wenn das Spiel gestartet wird. Ebenso kann die administrierende Person die erstellten Spiele bearbeiten, um Änderungen vorzunehmen oder Fehler zu korrigieren. Um ein Spiel zu starten, muss die administrierende Person auf die Schaltfläche „Präsentieren“ klicken, um das Spiel für die Zuschauenden zugänglich zu machen.

Verwaltung von Sponsoren und E-Mail-Texten

Neben der Verwaltung der Spiele soll der Administrationsbereich auch die Pflege von Sponsoren ermöglichen. Administrierende können Sponsoren erfassen, bearbeiten und löschen. Zu einem Sponsor gehören mindestens ein Name und ein Logo, das in der Präsentationsansicht auf dem Videowürfel angezeigt werden kann. Dadurch kann das Marketing Sponsoring-Elemente vorbereiten, ohne dass dafür eine Anpassung am Quellcode notwendig ist.

Zusätzlich soll pro Sponsor ein E-Mail-Text hinterlegt werden können. Dieser Text wird verwendet, wenn eine gewinnende Person ihre Kontaktdaten erfasst und eine Gewinnbenachrichtigung verschickt wird. Der E-Mail-Text kann dadurch an den jeweiligen Sponsor, den Gewinn oder die gewünschte Form der Abholung angepasst werden. So bleibt die Kommunikation nach dem Spiel flexibel und kann durch das Marketing selbst gepflegt werden.

Bei der Erstellung oder Bearbeitung eines Spiels kann anschliessend ein Sponsor ausgewählt werden. Das Spiel übernimmt dadurch sowohl die visuellen Sponsoring-Informationen für den Videowürfel als auch den passenden E-Mail-Text für die Gewinnabwicklung. Falls kein Sponsor ausgewählt ist, wird das Spiel ohne sponsorenspezifische Darstellung und ohne sponsorenspezifischen E-Mail-Text durchgeführt.

Starten und Stoppen von Spielen

In diesem Bereich können Spiele gestartet und gestoppt werden, um sie für die Zuschauenden zugänglich zu machen oder um sie zu beenden, wenn sie nicht mehr benötigt werden. Auch soll es möglich sein, die Fragen schneller zu wechseln, um bei Problemen das Quiz schneller, aber dennoch ordentlich zu beenden.

7.6 Mockups

7.6.1 Mockup-Zielsetzung

Die Mockups wurden mit Figma erstellt, um den Ablauf des Pausenspiels früh visuell zu prüfen. Dabei standen drei Ansichten im Fokus: die Darstellung auf dem Videowürfel, die mobile Teilnehmendenansicht und der Administrationsbereich. Die vollständige Sammlung der Entwurfsansichten befindet sich im Anhang A unter Mockups.

Die Entwurfsansichten zeigen, wie die Spielphasen Anmelden, Spielen, Antwortanzeige und Auswertung grundsätzlich aufgebaut sind. Da sich diese Phasen bei Quiz, Kreuzworträtsel und Bilderrätsel strukturell ähneln, werden im Hauptteil nur repräsentative Screens gezeigt.

Die erste Version der Anmeldephase auf dem Videowürfel ist dargestellt (siehe Abbildung 15).



Abbildung 15: Entwurfsansicht des Anmeldebildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma

Die zugehörige mobile Teilnehmendenansicht ist ebenfalls Teil der ersten Entwurfsvariante (siehe Abbildung 16).

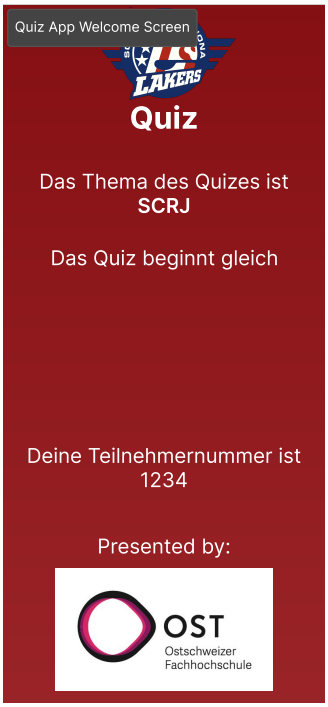


Abbildung 16: Entwurfsansicht des Anmeldebildschirms für alle Spielmodi in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma

Die geplante Hauptansicht des Administrationsbereichs ergänzt diese ersten Entwürfe (siehe Abbildung 17).

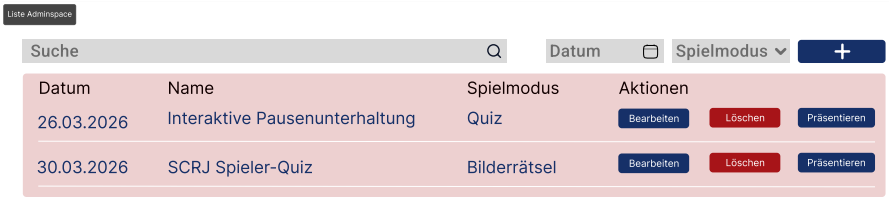


Abbildung 17: Entwurfsansicht der Hauptansicht des Administrationsbereichs
Quelle: Eigene Darstellung, erstellt mit Figma

7.6.2 Test im Stadion

Nach der Erstellung der Entwurfsansichten wurden diese im Stadion getestet, um Lesbarkeit, Flächennutzung und Verständlichkeit der Darstellung zu prüfen. Während des Tests wurden die Entwurfsansichten auf dem Videowürfel im Stadion angezeigt, um zu sehen, wie sie in der realen Umgebung wirken. Dabei wurden mehrere Annahmen aus der Gestaltung am Desktop relativiert. Im Entwurf wirkten Schriftgrössen, Logos und Abstände ausreichend gross, weil die Ansichten auf einem nahen Bildschirm betrachtet wurden. Auf dem Videowürfel veränderte sich diese Wahrnehmung durch die grössere Betrachtungsdistanz, die Höhe der Anzeige und die Ablenkung im Stadion. Deshalb mussten die Elemente stärker auf Fernwirkung und schnelle Erfassbarkeit ausgelegt werden.

Folgende Punkte sind während des Tests aufgefallen (siehe Abbildung 18):

- Auf dem Würfel können ganze Webseiten oder einzelne Bildschirmausschnitte dargestellt werden.
- Die Schriftgrösse auf dem Videowürfel sollte grösser sein, um besser lesbar zu sein, insbesondere für Zuschauende, die weiter entfernt sitzen. Die ursprüngliche Schriftgrösse war im Figma-Entwurf gut lesbar, erwies sich im Stadion aber als zu knapp, weil kurze Informationen aus grosser Distanz und innerhalb weniger Sekunden verstanden werden müssen.
- Das Logo wurde auf dem Videowürfel nicht priorisiert, da die verfügbare Fläche für Spielinhalt und Lesbarkeit benötigt wird und der Kontext im Stadion bereits durch die Umgebung gegeben ist. Die anfängliche Annahme, dass das Logo als zentrales Wiedererkennungselement prominent platziert werden sollte, wurde dadurch abgeschwächt. Für die Nutzung im Stadion sind Frage, Antwortmöglichkeiten, Timer und Einstiegshinweis wichtiger.
- Die Breite der Entwurfsansichten sollte auf die Breite des Videowürfels angepasst werden, damit die Anzeige nicht zu breit ist. Die ersten Entwurfsansichten orientierten sich stärker an einer allgemeinen Bildschirmdarstellung, da die genaue Auflösung des Videowürfels vor dem Test noch nicht bekannt war. Erst beim Test im Stadion wurde die konkrete Auflösung ermittelt.

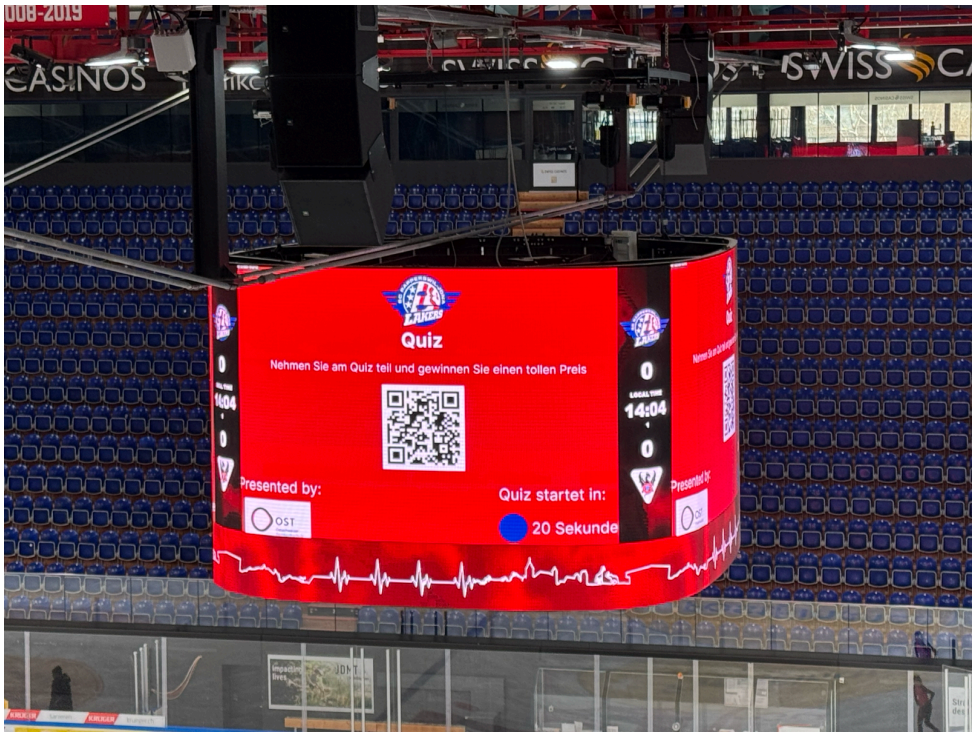


Abbildung 18: Foto vom Test der Entwurfsansichten im Stadion
Quelle: Eigene Aufnahme

7.6.3 Entwurfsansichten Version 2

Nach dem Test im Stadion wurden die Entwurfsansichten überarbeitet, um die oben genannten Punkte zu berücksichtigen und eine bessere Nutzungserlebnis zu bieten. Die Schriftgrösse wurde vergrössert, das Logo wurde entfernt und die Breite der Entwurfsansichten wurde angepasst, um besser auf den Videowürfel zu passen.

Die überarbeitete Anmeldephase zeigt die Anpassungen an die Stadiondarstellung (siehe Abbildung 19).



Abbildung 19: Überarbeitete Entwurfsansicht des Anmeldebildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma

Die überarbeitete Frageanzeige für das Kreuzworträtsel wurde ebenfalls auf bessere Fernwirkung ausgelegt (siehe Abbildung 20).



Abbildung 20: Überarbeitete Entwurfsansicht des Fragebildschirms für den Spielmodus Kreuzworträtsel
Quelle: Eigene Darstellung, erstellt mit Figma

Die überarbeitete Auswertungsphase schliesst die zweite Entwurfsiteration ab (siehe Abbildung 21).



Abbildung 21: Überarbeitete Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma

8 Systemkonzept und Architektur

Dieses Kapitel beschreibt das technische Systemkonzept und die Architektur der Anwendung. Der Fokus liegt auf den zentralen Entwurfsentscheidungen, den eingesetzten Technologien und dem Zusammenspiel der Systembestandteile.

8.1 Ziel der Umsetzung

Ziel der Umsetzung war es, aus dem zuvor erarbeiteten Konzept eine lauffähige Anwendung für interaktive Pausenspiele im Stadion der Rapperswil-Jona Lakers zu entwickeln. Die Anwendung sollte es ermöglichen, vorbereitete Spiele auf dem Videowürfel zu präsentieren und Zuschauende über ihre Mobilgeräte aktiv daran teilnehmen zu lassen. Damit wurde der Fokus nicht nur auf die Darstellung einzelner Spielmodi gelegt, sondern auf eine vollständige technische Lösung mit Frontend, Backend, Datenhaltung, Administration und Deployment.

Ein zentrales Ziel war die Trennung der verschiedenen Nutzungssituationen. Für das Publikum wurde eine mobile Teilnehmendenansicht umgesetzt, über die Teilnehmende einem laufenden Spiel beitreten, Fragen beantworten und am Ende eine Rückmeldung zum Ergebnis erhalten können. Für den Videowürfel wurde eine separate Präsentationsansicht realisiert, welche die Spielinhalte, den aktuellen Zustand, Timer, Antworten und Sponsoring-Elemente grossflächig darstellt. Für Administrierende wurde ein Administrationsbereich umgesetzt, in dem Spiele vorbereitet, bearbeitet, gestartet und ausgewertet werden können.

Die Umsetzung umfasst die drei im Konzept definierten Spielmodi Quiz, Kreuzworträtsel und Bilderrätsel. Alle Modi folgen demselben grundlegenden Ablauf: Ein Spiel wird im Administrationsbereich erstellt, mit Fragen und Antworten konfiguriert, gestartet und anschliessend synchron auf Präsentationsansicht und Mobilgeräten gespielt. Beim Quiz werden Antwortoptionen gewählt, beim Kreuzworträtsel werden Textantworten eingetragen und beim Bilderrätsel werden Bildinhalte mit Filtereffekten kombiniert.

Neben den Spielfunktionen war auch die technische Betreibbarkeit Teil des Umsetzungsziels. Dazu gehören eine persistente Speicherung der Spiel- und Ergebnisdaten, eine API für die Kommunikation zwischen Frontend und Backend, Echtzeitkommunikation während laufender Spiele sowie ein Deployment, das die Anwendung ausserhalb der lokalen Entwicklungsumgebung nutzbar macht. Zusätzlich wurden Administrations- und Sicherheitsaspekte berücksichtigt, damit Spielinhalte nicht öffentlich verändert werden können und die Anwendung in einer produktionsnahen Umgebung betrieben werden kann.

Am Ende der Umsetzung liegt damit ein funktionsfähiger MVP vor, der den gesamten Ablauf eines interaktiven Stadion-Spiels abbildet: von der Erstellung eines Spiels über die Live-Durchführung bis zur Ermittlung und Erfassung einer gewinnenden Person. Die Lösung bildet zugleich eine technische Grundlage, auf der eine spätere Integration in die bestehende Lakers-App oder eine Erweiterung um zusätzliche Spielmodi aufgebaut werden kann.

8.2 Übersicht der verwendeten Technologien

Die folgende Übersicht zeigt die wichtigsten Technologien, die für Entwicklung, Betrieb und Qualitätssicherung der Anwendung eingesetzt werden. Dabei werden sowohl die zentralen Bestandteile von Frontend, Backend und Datenbank als auch unterstützende Werkzeuge für Build, Testen und Deployment berücksichtigt.

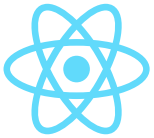
8.2.1 Programmiersprachen

Die Implementierung basiert hauptsächlich auf TypeScript und JavaScript. TypeScript wird vor allem im Backend und für typisierte Frontend-Module eingesetzt, während JavaScript weiterhin für einzelne Frontend-Einstiegspunkte verwendet wird. CSS definiert das visuelle Erscheinungsbild der Bedienoberfläche. Konfigurations- und Automatisierungsaufgaben werden ergänzend mit YAML und Shell-Skripten abgebildet.



8.2.2 React und Vite

Für den Frontend-Bereich der Applikation wurde React gewählt. React setzte sich gegenüber anderen Frameworks durch, da die einfache Struktur bereits aus vergangenen Projekten bekannt war. Ein weiterer Vorteil von React liegt in der komponentenbasierten Architektur. Dadurch können einmal erstellte Komponenten mehrfach wiederverwendet werden, wodurch die Entwicklung effizienter und übersichtlicher wird. Der lokale Entwicklungsserver und der Produktionsbuild werden mit Vite umgesetzt. Für die Navigation innerhalb der Anwendung wird React Router verwendet. QR-Codes für den Spielbeitritt werden mit `qrcode.react` dargestellt.



8.2.3 Node.js und NestJS

Für das Backend der Applikation wurde Node.js gewählt. Node.js ermöglicht es, JavaScript sowohl im Frontend als auch im Backend zu verwenden, was die Entwicklung und Wartung der Anwendung erleichtert. Der Vorteil von Node.js in diesem Projekt liegt auch darin, dass beide Teammitglieder bereits Erfahrungen mit Node.js machen konnten. Auf Node.js wird NestJS eingesetzt. NestJS strukturiert die Backend-Anwendung in Module, Controller, Services und Gateways. Dadurch lassen sich REST-Endpunkte, WebSocket-Kommunikation und Datenbankzugriffe sauber voneinander trennen. Für die Echtzeitkommunikation wird Socket.IO verwendet, damit Spielzustände während eines laufenden Spiels an die verbundenen Clients übertragen werden können.



8.2.4 PostgreSQL und TypeORM

Für die Datenbank der Applikation wurde PostgreSQL gewählt. PostgreSQL ist ein leistungsstarkes, objektrelationales Datenbankmanagementsystem, das sich durch seine Stabilität, Skalierbarkeit und umfangreichen Funktionen auszeichnet. Es bietet eine breite



Palette von Datentypen, unterstützt komplexe Abfragen und Transaktionen und verfügt über eine aktive Community, die kontinuierlich Verbesserungen und Erweiterungen bereitstellt. Im Projekt ermöglicht PostgreSQL eine effiziente Speicherung und Verwaltung der Daten, die von der Anwendung generiert werden. Die Persistenzschicht wird mit TypeORM umgesetzt. Dadurch können Datenbanktabellen als Entities im Code modelliert und Migrationen verwaltet werden.

8.2.5 Docker, Docker Compose und Nginx

Um das Frontend, das Backend und die Datenbank auf dem Server bereitzustellen, wird Docker eingesetzt. Docker ist eine weit verbreitete Plattform zur Containerisierung von Anwendungen. Im Gegensatz zur klassischen Virtualisierung virtualisiert Docker keine ganzen Betriebssysteme, sondern isoliert Anwendungen leichtgewichtig in Containern. Dies ermöglicht es, die Ressourcen pro Service gezielt zu steuern und die Auslastung des Servers effizient zu optimieren. Für die lokale Ausführung werden Frontend, Backend und PostgreSQL mit Docker Compose orchestriert. Der Frontend-Build wird im produktiven Container über Nginx ausgeliefert.



8.2.6 Swagger und OpenAPI

Für die Dokumentation der API wird Swagger verwendet. Swagger ist in der Industrie weit verbreitet, um APIs zu dokumentieren. Im Backend wird die Dokumentation über `@nestjs/swagger` und `swagger-ui-express` generiert. Dadurch können die REST-Endpunkte direkt geprüft und als Schnittstellendokumentation verwendet werden. Die Schnittstelle wird in dem Kapitel 8.5 genauer erläutert.



8.2.7 GitLab CI/CD und AWS

Für die Versionsverwaltung und Zusammenarbeit im Team wird GitLab verwendet. GitLab bietet nicht nur zentrale Repositorys, sondern unterstützt den Entwicklungsprozess durch integrierte Funktionen wie Merge Requests oder Code Reviews. Zusätzlich wird die CI/CD-Pipeline von GitLab genutzt, um die Anwendung und die Dokumentation automatisiert zu builden, zu testen und zu deployen. Dadurch können Fehler frühzeitig erkannt und neue Änderungen zuverlässig sowie konsistent ausgeliefert werden. Für das Hosting der gesamten Anwendung wird AWS eingesetzt. Die konkrete Infrastruktur und das Deployment werden im Kapitel 9.3 genauer beschrieben.



8.2.8 Weitere Entwicklungswerkzeuge und Bibliotheken

Als Paketmanager und Skript-Runner wird npm verwendet. Für die Codequalität werden ESLint und Prettier eingesetzt. Im Backend werden `class-validator` und `class-transformer` für die Validierung



und Transformation von DTOs verwendet. Die Konfiguration wird mit `@nestjs/config` und Joi validiert. Für Authentifizierung und geschützte Administrationsfunktionen wird JSON Web Token über `jsonwebtoken` eingesetzt. Dateiuploads werden über Multer verarbeitet. Für die Anbindung an die AWS-Infrastruktur wird das AWS SDK verwendet.

Im Backend werden Tests mit Jest, `ts-jest` und Supertest ausgeführt. Im Frontend werden Vitest, Testing Library und jsdom für Komponententests verwendet.

8.3 Systemarchitektur

Die Architektur ist als Webanwendung mit getrenntem Frontend, Backend und Persistenzschicht umgesetzt. Das Frontend stellt die Bedienoberflächen für den Administrationsbereich, die Präsentationsansicht und die mobile Teilnehmendenansicht bereit. Das Backend verarbeitet Fachlogik, Authentifizierung, Spielsteuerung, Datenzugriffe und die Kommunikation mit externen Diensten. PostgreSQL speichert die persistenten Anwendungsdaten. AWS-Dienste ergänzen die Anwendung für Dateiablage und E-Mail-Versand.

8.3.1 Gesamtaufbau

Die Anwendung besteht aus einer React-Anwendung, einem NestJS-Backend und einer PostgreSQL-Datenbank. Im lokalen Betrieb werden diese Bestandteile über Docker Compose ausgeführt. Im produktiven Betrieb werden Frontend und Backend als getrennte Container in ECR bereitgestellt. Der Frontend-Container liefert die gebaute React-Anwendung über Nginx aus. Nginx leitet Anfragen an `/api` und `/socket.io` an das Backend weiter, wo diese verarbeitet werden.

Diese Struktur trennt Darstellung von Fachlogik und Datenhaltung. Alle fachlichen Operationen laufen über das Backend. Dadurch bleiben Validierung, Zugriffsschutz und Spielsteuerung zentral kontrollierbar.

8.3.2 Frontend

Single-Page-Anwendung und Routing

Das Frontend ist als Single-Page-Anwendung mit React und React Router umgesetzt. Diese Entscheidung passt zum Ablauf des Pausenspiels, weil die Anwendung mehrere Ansichten mit gemeinsamem Zustand bereitstellt, aber keine serverseitig gerenderten Seiten benötigt. Eine Alternative wären getrennte Frontends für Administrationsbereich, Videowürfel und mobile Teilnahme gewesen. Dadurch wären die Einstiegspunkte stärker isoliert, gemeinsame Komponenten, API-Zugriffe und Spielzustandslogik müssten jedoch mehrfach gepflegt oder zusätzlich geteilt werden. Die Umsetzung als eine React-Anwendung reduziert diese Duplikation und erlaubt trotzdem getrennte Routen für die verschiedenen Nutzungskontexte.

Sie unterscheidet zwischen administrativen Ansichten, Präsentationsansicht und mobiler Teilnehmendenansicht. Der Administrationsbereich liegt unter `/adminspace`. Die Präsentationsansicht für den Videowürfel wird über `/present/:runId` geöffnet. Die mobile Teilnehmendenansicht läuft über `/play/:runId`. Die Routen wurden bewusst nach Nutzungskontext getrennt. Damit kann die Spielleitung den Administrationsbereich verwenden, während Publikum und Videowürfel über öffentliche, durch den Durchlauf identifizierte URLs auf denselben Spielzustand zugreifen. Die `runId` in den öffentlichen Routen vermeidet zusätzliche Navigation und erlaubt es, einen konkreten Durchlauf direkt per QR-Code oder Link zu öffnen.

Strukturierung des Frontends

Die Frontend-Struktur folgt einer Trennung zwischen Seiten, wiederverwendbaren Komponenten, Hooks, Services, Types und Utils. Die Seitenkomponenten sind in Kapitel 9.1 beschrieben und bilden die fachlichen Ansichten. Komponenten kapseln wiederverwendbare UI-Elemente wie Tabellen, Formulare, Eingabefelder oder spielmodusabhängige Darstellungen. Hooks bündeln zustandsbezogene Logik, beispielsweise für Listen, Formulare, Countdownanzeigen oder mobile Spielsitzungen. Services kapseln den Zugriff auf die REST-Programmierschnittstelle und stellen der Oberfläche fachlich benannte Funktionen bereit. Diese Trennung wurde gewählt, damit fachliche Ansichten nicht gleichzeitig Routing, Datenzugriff, Zustandshaltung und Darstellung enthalten. Eine flachere Struktur mit der gesamten Logik direkt in den Seitenkomponenten wäre zu Beginn einfacher gewesen, hätte aber bei drei Spielmodi und drei Ansichten schnell zu Wiederholungen geführt. Durch Hooks und Services kann gemeinsame Logik wiederverwendet werden, ohne dass die Bedienoberflächen stärker an konkrete API-Endpunkte gekoppelt werden.

Trennung von öffentlichen und geschützten Ansichten

Die Trennung der Frontend-Komponenten und Routen ist dargestellt (siehe Abbildung 22).

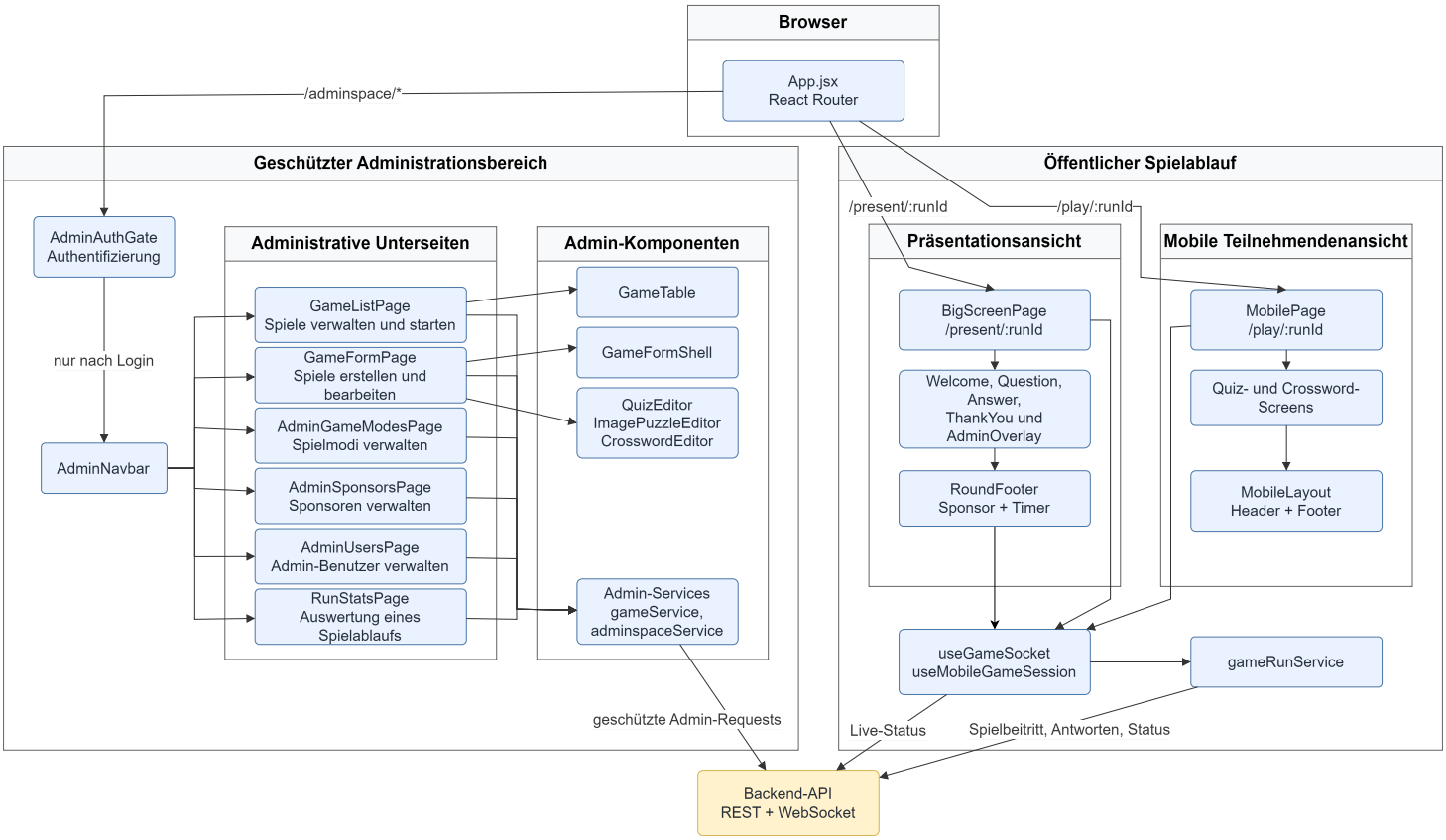


Abbildung 22: Komponentendiagramm des Frontends

Der Administrationsbereich ist durch eine Authentifizierungskomponente geschützt. Nur nach erfolgreicher Anmeldung werden die administrativen Unterseiten angezeigt. Die mobile Teilnehmendenansicht und die Präsentationsansicht sind davon getrennt, da sie für den öffentlichen Spielablauf vorgesehen sind. Diese Trennung reduziert die Einstiegshürde für Zuschauende, da für die Teilnahme keine Anmeldung erforderlich ist. Gleichzeitig bleiben administrative Funktionen wie das Erstellen, Bearbeiten und Starten von Spielen vom öffentlichen Ablauf getrennt.

8.3.3 Zusammenspiel der Ansichten

Während eines Spiels arbeiten Administrationsbereich, Präsentationsansicht und mobile Teilnehmendenansicht auf demselben Durchlauf. Der Administrationsbereich erstellt und startet einen Durchlauf. Die mobile Teilnehmendenansicht ermöglicht den Beitritt und die Antwortabgabe. Die Präsentationsansicht zeigt den aktuellen Spielzustand für das Publikum.

Das Backend ist die zentrale Instanz für den Spielzustand. Es verarbeitet administrative Aktionen und Eingaben der Teilnehmenden. Anschliessend verteilt es den aktuellen Zustand an die verbundenen Ansichten. Dadurch verwenden alle Clients dieselbe fachliche Grundlage. Die konkrete Kommunikation zwischen den Komponenten wird im Kapitel 8.5 beschrieben.

8.3.4 Backend

Das Backend ist mit NestJS umgesetzt. Das zentrale Modul ist `AppModule`. Es bindet Konfiguration, Datenbankanbindung, Speicheranbindung, Administrationsbereich und Spielausführung ein.

Die fachliche Backend-Struktur ist in Module gegliedert. Das `AdminSpaceModule` enthält die Verwaltung von Administrationskonten, Sponsoren, Spielmodi, Spielvorlagen, Fragen und Antworten. Es stellt Controller für HTTP-Anfragen bereit und verwendet Services für die fachliche Verarbeitung. Der `AdminSpaceController` nimmt die Anfragen zur Inhaltspflege entgegen und übergibt sie an den `AdminSpaceService`. Dieser validiert die fachlichen Zusammenhänge, lädt benötigte Referenzen und koordiniert die Speicherung über TypeORM. Das `GameRunModule` enthält die Durchführung laufender Spiele. Dazu gehören die Steuerung von Durchläufen, die Teilnahme von Zuschauenden, die Antwortverarbeitung, die Auswertung und die Echtzeitverteilung des Spielzustands.

Das `StorageModule` kapselt die Dateiablage. Es stellt Funktionen für Bilddateien bereit, ohne dass die Fachmodule direkt mit dem externen Objektspeicher arbeiten müssen. Das `MailModule` kapselt den Versand von Gewinnbenachrichtigungen. Diese Modulstruktur reduziert die Kopplung zwischen Fachlogik und Infrastruktur. Die wichtigsten Backend-Klassen und ihre Beziehungen sind dargestellt (siehe Abbildung 23).

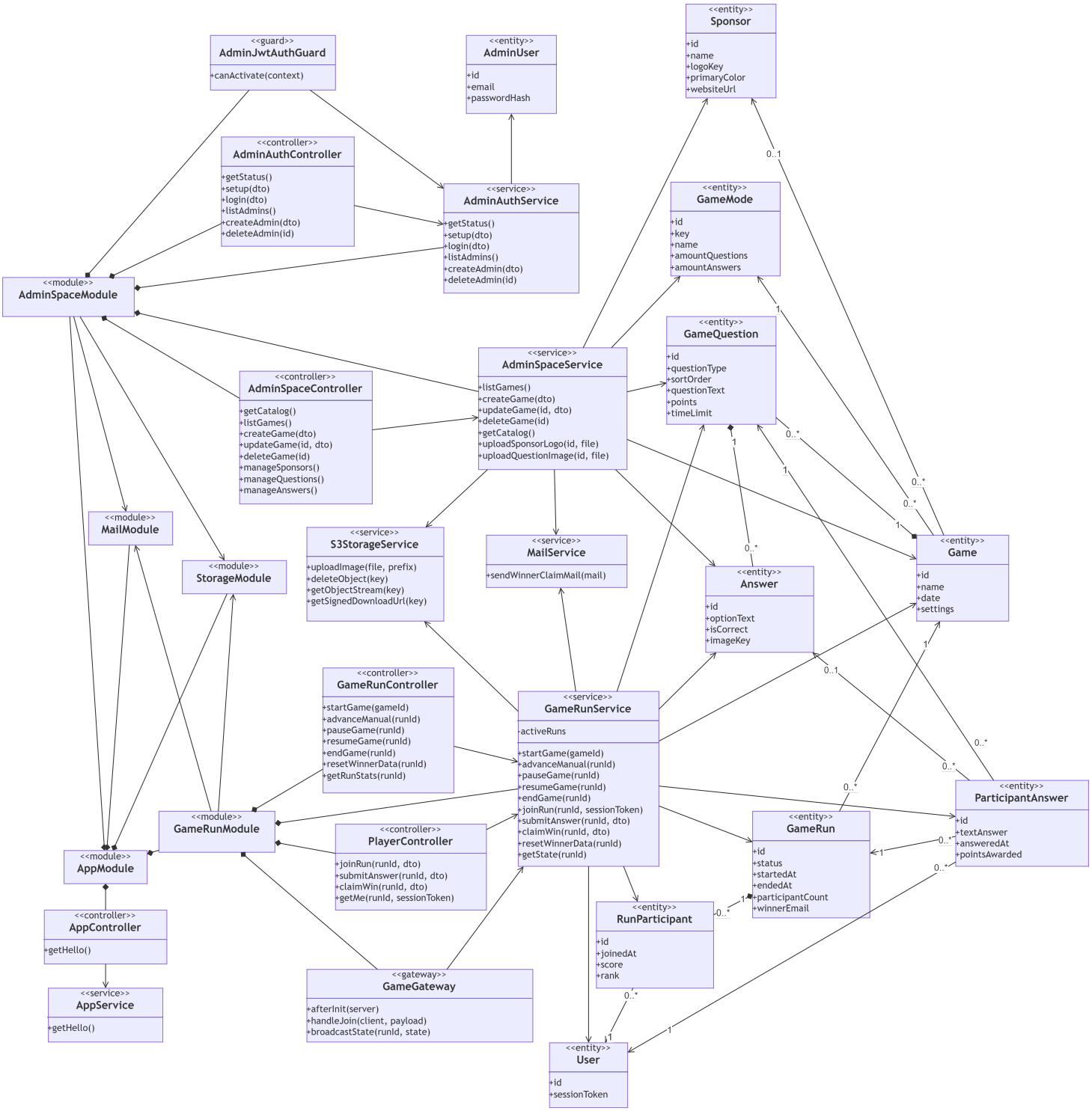


Abbildung 23: Klassendiagramm des Backends

8.3.5 Datenbank und Persistenz

Die Persistenzschicht basiert auf PostgreSQL und TypeORM. Das Backend verwendet Entitäten und Repositories, um Datenbanktabellen fachlich abzubilden. Die Datenbank enthält vorbereitete Spielinhalte, Laufzeitdaten, Teilnahmen, Antworten, Sponsoren und Administrationskonten. Die detaillierte Tabellenstruktur wird im Kapitel 8.4 beschrieben.

Die Datenbank ist keine direkte Schnittstelle des Frontends. Frontend-Anfragen werden zuerst durch Controller und Services im Backend verarbeitet. Erst danach erfolgen Lese- oder Schreibzugriffe über TypeORM. Diese Architektur stellt sicher, dass fachliche Regeln und Validierung vor der Persistierung angewendet werden.

8.3.6 Externe Dienste und Infrastruktur

Für die produktive Umgebung werden Dienste von AWS verwendet. Der Betrieb der Container erfolgt über Amazon Elastic Container Service. Die produktive PostgreSQL-Datenbank wird über Amazon Relational Database Service bereitgestellt. Bilddateien werden in Amazon Simple Storage Service abgelegt. Gewinnbenachrichtigungen können über Amazon Simple Email Service versendet werden. Sensible Konfigurationswerte werden über AWS Systems Manager Parameter Store eingebunden. Die konkrete AWS-Architektur und das Deployment werden in Kapitel 9.3, insbesondere in Kapitel 9.3.4, beschrieben.

Diese externen Dienste sind im Backend oder in der Betriebsumgebung gekapselt. Die Bedienoberflächen greifen nicht direkt auf diese Dienste zu. Sie kommunizieren mit dem Backend, welche alle Zugriffe koordiniert.

8.4 Datenmodell und Persistenz

Die Anwendung verwendet PostgreSQL als relationale Datenbank und TypeORM als Persistenzschicht im NestJS-Backend. Das Datenmodell ist fachlich in zwei Bereiche gegliedert, die beide in derselben Datenbank liegen und miteinander verknüpft sind. Der erste Bereich umfasst vorbereitete Stammdaten wie Spielvorlagen, Fragen, Antworten, Spielmodi und Sponsoren. Der zweite Bereich umfasst Laufzeitdaten, die erst während eines gestarteten Spiels entstehen, beispielsweise Durchläufe, Teilnehmende und abgegebene Antworten. Diese Trennung verhindert, dass vorbereitete Inhalte durch eine konkrete Durchführung verändert werden, und erlaubt gleichzeitig die spätere Auswertung einzelner Spielrunden. Ein relationales Modell passt zu diesem Anwendungsfall, da Spiele, Fragen, Antworten, Durchläufe und Teilnehmende explizite Beziehungen zueinander besitzen.

Das Diagramm zeigt die grundlegende Struktur der persistenten Daten und alle Beziehungen zwischen Spielvorlagen, Spielausführungen und Teilnehmenden. Die Tabellen verwenden jeweils eine technische `id` als Primärschlüssel. Die Verbindungslinien zeigen die Foreign-Key-Beziehungen zwischen den Tabellen. Die nachfolgenden Abschnitte beschreiben die fachlichen Bereiche des Modells und ergänzen die relevanten Umsetzungsdetails (siehe Abbildung 24).

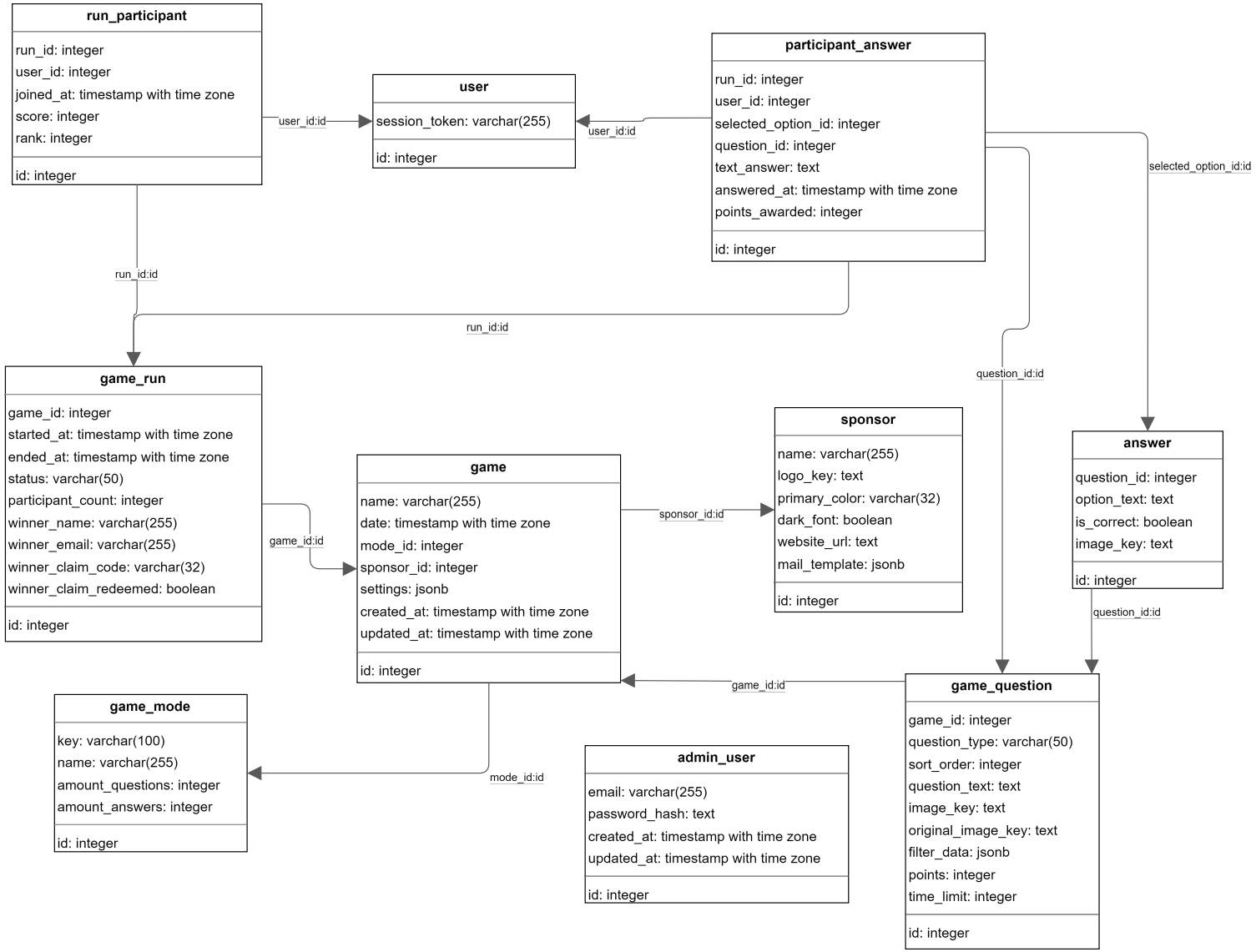


Abbildung 24: Datenbankstruktur
Quelle: Eigene Abbildung, erstellt mit draw.io

8.4.1 Spielvorlagen und Stammdaten

Die Tabelle `game` bildet eine vorbereitete Spielvorlage ab. Sie enthält den Namen, das geplante Datum, den gewählten Spielmodus, optionale Sponsorendaten sowie zusätzliche Einstellungen im Feld `settings`. Dieses Feld ist als JSONB gespeichert, damit spielmodusabhängige Konfigurationen abgelegt werden können, ohne für jede Variante neue Tabellen erstellen zu müssen. Ein Beispiel dafür sind zusätzliche Parameter für das Kreuzworträtsel, die nicht für jeden Spielmodus relevant sind. Ein Spiel besitzt mehrere Einträge in `game_question`. Die Reihenfolge der Fragen wird über `sort_order` festgelegt.

`game_mode` beschreibt die unterstützten Spieltypen. Aktuell werden die Modi Quiz, Bilderquiz und Kreuzworträtsel initial angelegt. Neben einem eindeutigen Schlüssel speichert die Tabelle auch die erwartete Anzahl an Fragen und Antworten. Dadurch kann das Frontend Formulare und Validierungen anhand der gewählten Spielart aufbauen.

`game_question` enthält die eigentliche Aufgabenstellung. Alle Spielmodi verwenden dieselbe Fragentabelle, unterscheiden sich aber über das Feld `question_type`. Für klassische Quizfragen werden Antwortoptionen in der Tabelle `answer` gespeichert. Für das Bilderquiz können zusätzlich Bildreferenzen sowie Filterinformationen abgelegt werden. Diese Filterdaten werden ebenfalls als JSONB gespeichert, weil sie eine flexible Struktur besitzen und sich je nach Bildbearbeitung unterscheiden können. Dadurch können neue Filterparameter ergänzt werden, ohne das relationale Schema bei jeder Anpassung erweitern zu müssen. Beim Kreuzworträtsel wird die gesuchte Lösung als Antwort gespeichert, während die Frage den Hinweistext enthält.

Die Tabelle `sponsor` enthält Branding-Informationen wie Name, Logo, Primärfarbe, Schriftkontrast, E-Mailnachricht und Webseite. Ein Spiel kann optional einem Sponsor zugeordnet werden. Bild- und Logodateien werden nicht direkt in der Datenbank gespeichert. Felder wie `logo_key`, `image_key` und `original_image_key` enthalten lediglich Referenzen auf die Dateien im Storage. Damit bleibt die Datenbank auf strukturierte Metadaten beschränkt, während grössere Binärdaten separat abgelegt werden.

8.4.2 Laufzeitdaten

Sobald ein Spiel gestartet wird, erzeugt das Backend einen Eintrag in `game_run`. Dieser Datensatz verweist auf die zugrunde liegende Spielvorlage und enthält den Status des Durchlaufs, Start- und Endzeitpunkt, Anzahl Teilnehmende sowie optionale Gewinninformationen. Die eigentliche Steuerung des laufenden Spiels erfolgt im Backend, die relevanten Resultate werden jedoch persistent gespeichert. Die Datenbank enthält damit den nachvollziehbaren Zustand und die Ergebnisse eines Durchlaufs, aber kein vollständiges Event-Log jeder Live-Aktion.

Teilnehmende werden über die Tabelle `user` abgebildet. Da für das Publikum keine Registrierung vorgesehen ist, wird ein anonymer Session-Token verwendet. Die Verbindung zwischen einem User und einem konkreten Durchlauf wird in `run_participant` gespeichert. Dort werden Beitrittszeitpunkt, Punktestand und Rang festgehalten. Ein erneuter Beitritt mit demselben Session-Token erzeugt keine zusätzliche Teilnahme. Dadurch bleiben die Zuordnung zum Durchlauf und der bisherige Fortschritt auch nach einem Neuladen der mobilen Teilnehmendenansicht erhalten.

Die abgegebenen Antworten werden in `participant_answer` gespeichert. Je nach Spielmodus verweist ein Eintrag entweder auf eine ausgewählte Antwortoption oder enthält eine freie Texteingabe. Zusätzlich werden Antwortzeitpunkt und vergebene Punkte persistiert.

8.4.3 Administration

Für die Administration existiert die Tabelle `admin_user`. Sie speichert die E-Mail-Adresse und den gehashten Passwortwert der Administrationskonten. Passwörter werden dadurch nicht im Klartext persistiert. Die Tabelle ist bewusst nicht mit den fachlichen Spieldaten verbunden, da sie ausschliesslich für die Authentifizierung im Administrationsbereich verwendet wird. Die Administration greift über die TypeORM-Repositories auf die Tabellen für Spiele, Fragen, Antworten, Spielmodi und Sponsoren zu. Beim Erstellen oder Bearbeiten eines Spiels werden zusammengehörige Änderungen, beispielsweise Spielinformationen, Fragen und Antworten, transaktional gespeichert. Die Transaktion wird nur vollständig übernommen, wenn alle Teilschritte erfolgreich sind. Tritt während der Speicherung ein Fehler auf, bleibt der bisherige Datenbestand unverändert, sodass keine unvollständige Spielvorlage entsteht.

8.4.4 Validierung und Datenintegrität

Die Datenintegrität wird auf mehreren Ebenen sichergestellt. Eingaben aus dem Frontend werden im Backend über DTOs validiert, bevor sie in Entitäten überführt und gespeichert werden. Dadurch werden ungültige oder unvollständige Daten bereits an der Schnittstelle abgefangen. Zusätzlich erzwingen Datenbank-Constraints die referenzielle Integrität zwischen den Tabellen. Beispiele dafür sind eindeutige Schlüssel wie `game_mode.key`, die eindeutige Reihenfolge von Fragen innerhalb eines Spiels, die einmalige Teilnahme pro Durchlauf und die Einschränkung, dass eine Person dieselbe Frage in einem Durchlauf nur einmal beantworten kann. Fachliche Regeln, etwa die erlaubte Anzahl Antworten pro Spielmodus oder das Berechnen der Punkte, werden in den Services des Backends umgesetzt.

8.4.5 Beziehungen und Löschverhalten

Die Beziehungen im Datenmodell folgen den fachlichen Abhängigkeiten. Fragen und Antworten gehören direkt zu einem Spiel und werden beim Löschen des Spiels mitentfernt. Das gleiche gilt für Durchläufe, Teilnahmen und abgegebene Antworten, da diese ohne ihre übergeordneten Daten keinen eigenständigen fachlichen Wert haben. Bei Spielmodi wird dagegen `ON DELETE RESTRICT` verwendet, damit ein Spielmodus nicht gelöscht werden kann, solange noch Spiele darauf verweisen. Bei Sponsoren wird `ON DELETE SET NULL` eingesetzt, weil Sponsoring-Informationen optional sind.

Für häufig verwendete Fremdschlüssel wurden Indizes angelegt. Das betrifft unter anderem die Zuordnung von Spielen zu Spielmodi und Sponsoren, Fragen zu Spielen, Antworten zu Fragen sowie Laufzeitdaten zu Durchläufen und Teilnehmenden. Diese Indizes unterstützen insbesondere die Listenansichten im Administrationsbereich und die Abfragen während eines laufenden Spiels.

8.4.6 Datenschutzrelevante Speicherung

Für die Teilnahme am Spiel ist keine Registrierung erforderlich. Personenbezogene Daten werden nur dort gespeichert, wo sie für den Ablauf notwendig sind. Dies betrifft insbesondere optionale Gewinninformationen wie Name und E-Mail-Adresse, falls nach einem Durchlauf ein Gewinn abgewickelt werden soll. Eine separate Funktion erlaubt es, diese Informationen bei Bedarf nach der Abwicklung wieder aus dem Durchlauf zu entfernen, um die Einhaltung der Datenschutzvorgaben zu gewährleisten (Bundesgesetz über den Datenschutz (Datenschutzgesetz, DSG), SR 235.1, 2022).

8.4.7 Migrationen und Schema-Verwaltung

Das Datenbankschema wird über TypeORM-Migrationen verwaltet. Die Synchronisation von TypeORM ist deaktiviert, damit Schemaänderungen nicht automatisch und unkontrolliert auf die Datenbank angewendet werden. Stattdessen definieren Migrationen explizit, welche Tabellen, Fremdschlüssel, Indizes und Seed-Daten erstellt werden. Die initiale Migration legt das vollständige Schema an und befüllt die Datenbank mit den grundlegenden Spielmodi, Sponsoren und Beispielspielen.

Die Datenbankverbindung wird über Umgebungsvariablen konfiguriert. Dazu gehören Host, Port, Benutzername, Passwort und Datenbankname. Für die produktive Umgebungen ist zusätzlich SSL aktiviert und ein CA-Zertifikat wurde hinterlegt. Beim Deployment entscheidet die Konfiguration, ob Migrationen beim Start des Backends automatisch ausgeführt werden.

8.5 Schnittstellen

Dieses Unterkapitel beschreibt die Schnittstellen der Anwendung. Der Begriff Schnittstelle umfasst dabei die Kommunikation zwischen Frontend und Backend, die Echtzeitkommunikation während eines laufenden Spiels sowie die technischen Anbindungen an Datenbank und Clouddienste. Die Details dazu sind in den Unterkapiteln Datenmodell, Sicherheitsmassnahmen und Technische Herausforderungen beschrieben.

8.5.1 REST-API

Die REST-Programmierschnittstelle verbindet das Frontend mit dem NestJS-Backend. Alle REST-Endpunkte verwenden den Pfadpräfix `/api`. Als Austauschformat wird JavaScript Object Notation (JSON) verwendet. Für Bildübertragungen werden mehrteilige Formulardaten eingesetzt.

Die REST-Endpunkte sind fachlich in Administrations- und Teilnahmefunktionen gegliedert. Die Administrationsfunktionen liegen unterhalb von `/api/adminspace`. Sie dienen der Anmeldung, der Verwaltung von Sponsoren, Spielmodi, Spielvorlagen, Fragen und Antworten sowie der Steuerung laufender Durchläufe. Die Teilnahmefunktionen liegen unterhalb von `/api/player`. Sie ermöglichen den Beitritt zu einem Durchlauf, die Antwortabgabe, das Abfragen des eigenen Status und die Übermittlung von Gewinndaten.

8.5.2 Geschützter Administrationsbereich

Administrative Endpunkte sind geschützt, weil sie Spielinhalte und laufende Durchführungen verändern können. Die Anwendung verwendet dafür JSON Web Tokens. Nach erfolgreicher Anmeldung sendet der Administrationsbereich dieses Token bei geschützten Anfragen mit. Das Backend prüft das Token vor der Ausführung administrativer Aktionen.

Die Player-Endpunkte sind davon getrennt. Sie erlauben die Teilnahme an einem aktiven Spiel ohne Administrationskonto. Die serverseitige Trennung zwischen öffentlichen und geschützten Endpunkten verhindert, dass die mobile Teilnehmendenansicht administrative Funktionen auslösen kann.

8.5.3 Medien und Dateiübertragung

Bilddateien werden über REST-Endpunkte an das Backend übertragen. Dies betrifft Sponsorlogos und Fragenbilder. Das Backend legt die Dateien in Amazon Simple Storage Service ab und speichert in PostgreSQL nur die zugehörigen Objektschlüssel.

Für die Anzeige von Bildern stellt das Backend kontrollierte Zugriffe bereit. Der Client verwendet dadurch die Schnittstellen des Backends und greift nicht direkt auf den Objektspeicher zu.

8.5.4 Echtzeitkommunikation

Für den Betrieb verwendet die Anwendung Socket.IO. Die Verbindung läuft im Namensraum `/game`. Mobile Clients und Präsentationsansichten treten einem Raum bei, der einem konkreten Spieldurchlauf entspricht. Das Beitreten erfolgt über die Nachricht `join:run`. Zustandsänderungen werden über die Nachricht `game:state` verteilt.

Die Echtzeitkommunikation ergänzt die REST-Programmierschnittstelle. REST verarbeitet fachliche Aktionen wie Spielstart, Beitritt und Antwortabgabe. Socket.IO verteilt den daraus entstehenden Spielzustand an die verbundenen Clients. Das Backend bildet den aktuellen Zustand dafür in einem `GameStateDto` ab. Dieses enthält den Status und die Zeitangaben der laufenden Phase, die aktuelle Frage mit ihren Antwortoptionen, die Auflösung, Sponsorinformationen und gegebenenfalls die Gewinnnummer. Damit erhalten Administrationsbereich, Präsentationsansicht und mobile Teilnehmendenansicht einen einheitlichen Zustandssnapshot.

Auch die Zeitsteuerung bleibt serverseitig. Beim Pausieren übermittelt das Backend eine stabile Restzeit. Beim Fortsetzen berechnet es auf dieser Grundlage eine neue Endzeit für die laufende Phase. Die Clients zeigen die empfangenen Zeitwerte lediglich an und lösen selbst keine Phasenwechsel aus.

8.5.5 Datenbank

Das Backend greift über TypeORM auf PostgreSQL zu. Diese Schnittstelle wird ausschliesslich serverseitig verwendet. Das Frontend besitzt keinen direkten Zugriff auf die Datenbank. Die Datenbankverbindung wird über Umgebungsvariablen konfiguriert.

TypeORM bildet die Datenbanktabellen auf Entitäten im Backend ab. Die konkrete Tabellenstruktur, die Beziehungen und die Migrationen werden im Unterkapitel zum Datenmodell beschrieben.

8.5.6 AWS

Die AWS-Dienste sind externe technische Schnittstellen der Anwendung. Sie werden durch das Backend verwendet und erscheinen nicht direkt in den Bedienoberflächen. Die konkrete Dienstaufteilung ist in Kapitel 8.3 beschrieben. Das Deployment und der Betrieb werden in Kapitel 9.3 erläutert.

8.5.7 Swagger als API-Dokumentation

Swagger stellt die technische Referenz der REST-Endpunkte unter `/api/docs` bereit. Sie dokumentiert Pfade, HTTP-Methoden, Parameter, Anfrageformate und Antwortformate.

8.6 Sicherheitsaspekte

Dieses Unterkapitel beschreibt die wichtigsten Sicherheitsaspekte, die bei der Umsetzung der Anwendung berücksichtigt wurden. Da die Anwendung während eines Live-Spiels öffentlich erreichbar ist, wurde besonders darauf geachtet, administrative Funktionen von den öffentlichen Spielansichten zu trennen. Zuschauende sollen ohne Anmeldung an einem gestarteten Spiel teilnehmen können, dürfen aber keine Spielinhalte verändern oder administrative Aktionen auslösen.

8.6.1 Authentifizierung und geschützte Administrationsfunktionen

Der Administrationsbereich ist der sicherheitskritischste Bereich der Anwendung, da dort Spiele erstellt, bearbeitet, gestartet und gesteuert werden können. Der Zugriff auf diese Funktionen ist deshalb nicht öffentlich verfügbar. Für geschützte Backend-Endpunkte wird eine tokenbasierte Authentifizierung mit JSON Web Tokens verwendet. Nach erfolgreicher Anmeldung erhält der Client ein Token, das bei weiteren Anfragen mitgesendet wird. Das Backend prüft dieses Token, bevor administrative Aktionen ausgeführt werden. Der Ablauf geschützter administrativer Steueraktionen während einer laufenden Spielrunde ist im Sequenzdiagramm im Anhang dargestellt (siehe Abbildung 34).

Dadurch wird verhindert, dass unautorisierte Personen Spielinhalte ändern oder ein laufendes Spiel beeinflussen können. Die mobile Teilnehmendenansicht und die Präsentationsansicht greifen dagegen nur auf die Informationen zu, die für die Teilnahme oder Darstellung eines gestarteten Spiels notwendig sind. Diese Trennung unterstützt die Anforderungen an den beschränkten Zugriff auf den Administrationsbereich und reduziert die Angriffsfläche der öffentlichen Anwendung.

8.6.2 Autorisierung und Trennung der Anwendungsbereiche

Neben der reinen Anmeldung ist auch die fachliche Trennung der erlaubten Aktionen relevant. Administrative Aktionen wie das Erstellen von Spielen, das Bearbeiten von Fragen, das Starten eines Durchlaufs oder das manuelle Weiterschalten eines Spiels werden serverseitig geschützt. Die Bedienoberfläche blendet entsprechende Funktionen zwar nur im Administrationsbereich ein, die eigentliche Sicherheitsentscheidung liegt jedoch im Backend. Damit bleibt der Schutz auch dann wirksam, wenn eine Person versucht, API-Endpunkte direkt aufzurufen.

Die Spielteilnahme ist bewusst niederschwellig gehalten. Zuschauende gelangen über einen QR-Code zur mobilen Teilnehmendenansicht um einem aktiven Spiel beitreten. Dabei können sie nur Aktionen ausführen, die zum Spielablauf gehören, zum Beispiel das Abgeben einer Antwort. Nicht gestartete Spiele sind für Zuschauende nicht spielbar. Auch die Erfassung von Kontaktdaten für einen Gewinn wird serverseitig autorisiert. Bevor das Backend die Angaben speichert und einen Gewinncode erzeugt, prüft es, ob die Anfrage zur erstplatzierten Teilnahme des betreffenden Durchlaufs gehört.

8.6.3 Eingabevalidierung

Alle Eingaben aus dem Frontend werden im Backend validiert, bevor sie verarbeitet oder gespeichert werden. Dazu werden im NestJS-Backend Data Transfer Objects mit `class-validator` und `class-transformer` eingesetzt. Pflichtfelder, Datentypen und Wertebereiche können dadurch zentral beschrieben und geprüft werden. Dies betrifft sowohl administrative Formulare im Administrationsbereich als auch Eingaben von Teilnehmenden während eines Spiels.

Die Validierung ist besonders wichtig, weil Spielinhalte unterschiedliche Strukturen besitzen können. Quiz, Kreuzworträtsel und Bilderrätsel verwenden gemeinsame Grunddaten, enthalten aber auch spielmodusabhängige Felder. Durch die serverseitige Prüfung wird verhindert, dass unvollständige oder ungültige Daten einen laufenden Durchlauf stören. Zusätzlich werden Konfigurationswerte beim Start der Anwendung mit Joi geprüft. Fehlende oder fehlerhafte Umgebungsvariablen werden dadurch früh erkannt, statt erst während des Betriebs zu unerwartetem Verhalten zu führen.

8.6.4 Umgang mit sensiblen Daten

Sensible Informationen wie Datenbankpasswörter, JWT-Secrets oder Zugangsdaten zu externen Diensten werden nicht fest im Quellcode hinterlegt. Stattdessen werden sie über Umgebungsvariablen und Secret-Mechanismen der Betriebsumgebung bereitgestellt. Dies erleichtert die Trennung zwischen lokaler Entwicklung, Testumgebung und produktiver Umgebung.

8.6.5 Schutz der Live-Kommunikation

Für die Synchronisation zwischen Backend, Präsentationsansicht und mobiler Teilnehmendenansicht wird Socket.IO verwendet. Da über diese Verbindung der aktuelle Spielzustand verteilt wird, darf sie nicht als Steuerkanal für beliebige Client-Aktionen dienen. Clients erhalten die für ihre Ansicht notwendigen Zustandsinformationen, während administrative Steuerbefehle weiterhin den geschützten Administrationsfunktionen vorbehalten bleiben.

Das Backend verwaltet den massgeblichen Spielzustand zentral. Clients rekonstruieren den Ablauf nicht selbst, sondern erhalten serverseitig erzeugte Zustands-Snapshots. Dadurch können ungültige oder verspätete Client-Nachrichten begrenzt werden, ohne dass der gesamte Spielablauf von einzelnen Endgeräten abhängt.

8.6.6 Betrieb und weitere Schutzmassnahmen

Die Anwendung wird containerisiert betrieben. Frontend, Backend und Datenbank laufen als getrennte Dienste, wodurch Verantwortlichkeiten und Zugriffe klarer abgegrenzt werden können. Der Frontend-Build wird über Nginx ausgeliefert, während das Backend die API- und WebSocket-Kommunikation bereitstellt.

Im produktiven Betrieb geschehen externe Zugriffe über verschlüsselte Verbindungen. Zudem werden Debug-Ausgaben, interne Fehlermeldungen und sensible Konfigurationswerte nicht an Clients weitergegeben werden. Deshalb werden Fehler auf Anwendungsebene behandelt und in eine für Zuschauende geeignete Meldung übersetzt.

Die umgesetzten Sicherheitsmassnahmen konzentrieren sich auf die für den MVP wichtigsten Risiken: Schutz des Administrationsbereichs, Validierung von Eingaben, Trennung von öffentlichen und administrativen Funktionen sowie sicherer Umgang mit Konfigurationswerten. Weiterführende Massnahmen wie detaillierte Rollenmodelle, Audit-Logs oder erweitertes Rate Limiting können auf dieser Grundlage ergänzt werden, falls die Anwendung produktiv weitergeführt wird.

9 Realisierung und Betrieb

Dieses Kapitel beschreibt die konkrete Realisierung der Anwendung und ihre betriebliche Nutzung. Der Fokus liegt auf den umgesetzten Funktionen, den Bedienoberflächen, dem Deployment sowie der Qualitätssicherung.

9.1 Bedienoberfläche

Dieses Kapitel beschreibt die technische Umsetzung der Bedienoberflächen. Im Mittelpunkt stehen die realisierten Ansichten, die wiederverwendeten Oberflächenelemente, die spielmodusabhängige Formularlogik und das Verhalten auf unterschiedlichen Ausgabegeräten.

9.1.1 Administrationsbereich

Der Administrationsbereich dient der Verwaltung von Spielen, Spielmodi, Sponsoren und Administrationskonten. Nach der Anmeldung zeigt die Startansicht eine tabellarische Übersicht der vorhandenen Spielvorlagen. Suchfeld, Spielmodusfilter und Datumsfilter unterstützen das Auffinden vorbereiteter Spiele. Zusätzlich wird der Verlauf gestarteter Durchläufe dargestellt. Dadurch kann die Spielleitung erkennen, welche Durchläufe aktiv sind und welche Resultate bereits vorliegen. Die Startansicht des Administrationsbereichs zeigt diese Verwaltungsfunktionen (siehe Abbildung 25).

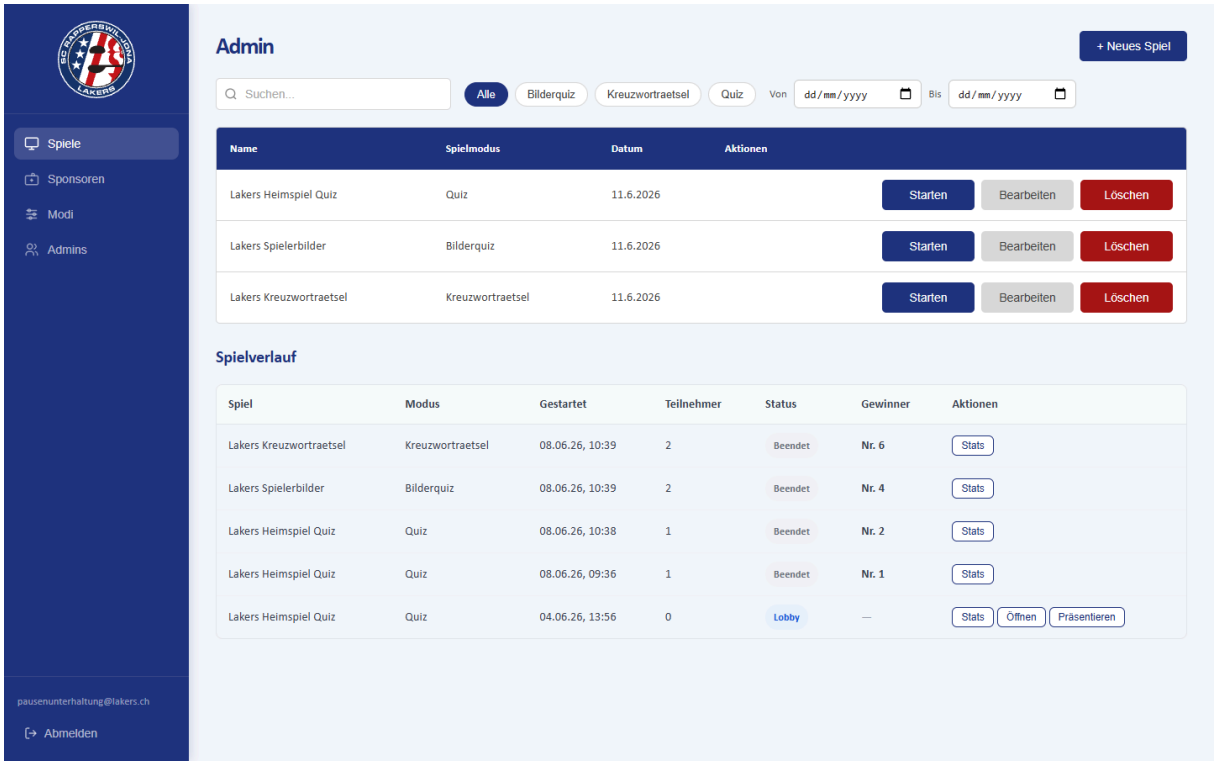


Abbildung 25: Übersicht des Administrationsbereichs mit Spielvorlagen und Spielverlauf
Quelle: Eigene Aufnahme

Spielvorlagen können gestartet, bearbeitet oder gelöscht werden. Aktive und abgeschlossene Durchläufe erscheinen im Spielverlauf. Dort können Detailstatistiken geöffnet oder Präsentationsansichten gestartet werden. Die Oberfläche unterstützt damit sowohl die Vorbereitung vor einem Spieltag als auch die Steuerung während einer Veranstaltung.

9.1.2 Formularlogik

Das Formular für Spielvorlagen ist in einen allgemeinen Formularbereich und einen Fragenbereich gegliedert. Der allgemeine Bereich enthält Name, Datum, Spielmodus und optionalen Sponsor. Der Fragenbereich passt seine Eingabefelder an den gewählten Spielmodus an. Bei einem Quiz werden Frage, Zeitlimit, Antwortoptionen und korrekte Lösung erfasst. Bei einem Bilderquiz ergänzt die Oberfläche die Erfassung um Bilddaten und Filter. Beim Kreuzworträtsel stehen Fragen und Lösung im Vordergrund. Ein Beispiel für die Bearbeitung einer Quizfrage zeigt die konkrete Formularstruktur (siehe Abbildung 26).

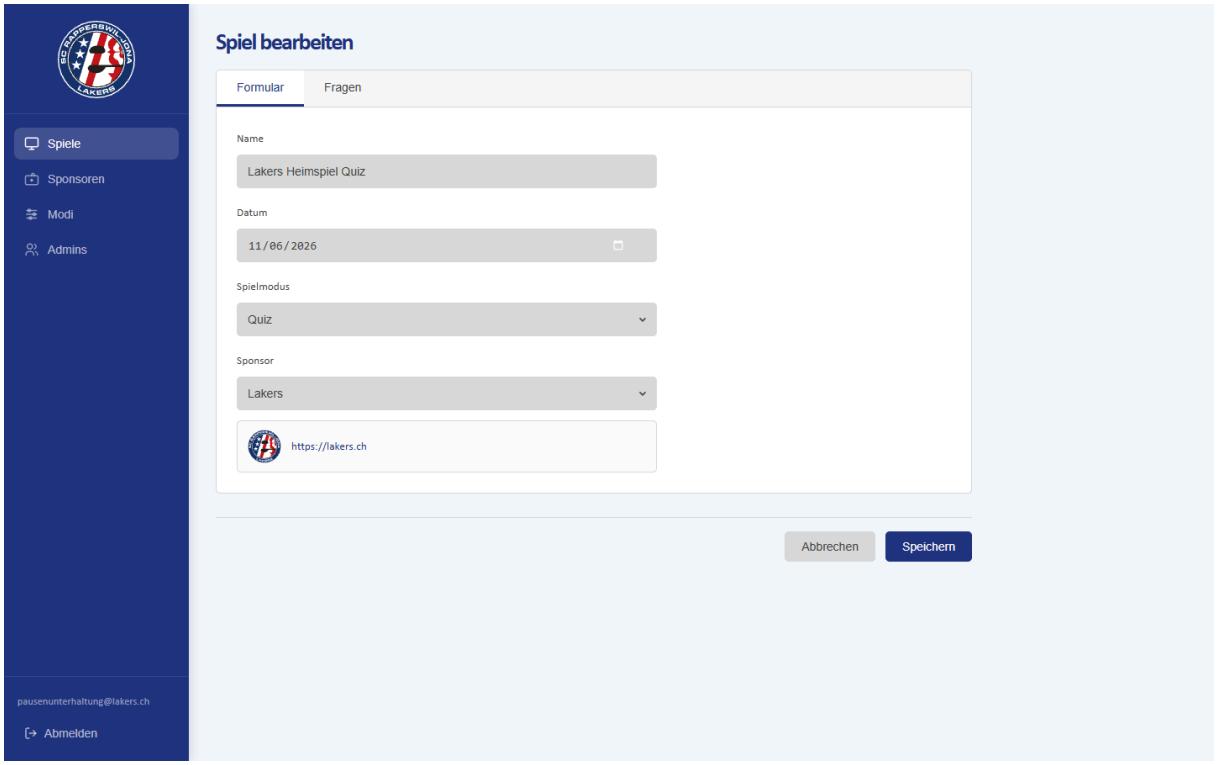


Abbildung 26: Bearbeitung einer Quizfrage im Administrationsbereich
Quelle: Eigene Aufnahme

Die komponentenbasierte Umsetzung reduziert Wiederholungen in der Oberfläche. Gemeinsame Elemente wie Eingabefelder, Schaltflächen, Tabellen, Filter und Fehlermeldungen werden wiederverwendet. Spielmodusabhängige Editoren kapseln die spezifische Eingabelogik. Dadurch bleibt die Formularstruktur einheitlich, obwohl die Inhalte der Spielmodi unterschiedlich sind.

9.1.3 Präsentationsansicht

Die Präsentationsansicht ist für die Anzeige auf dem Videowürfel vorgesehen. Sie verwendet eine reduzierte Darstellung mit grosser Schrift, hohem Kontrast und wenigen gleichzeitigen Informationen. Im Wartezustand zeigt sie den Spielnamen, eine kurze Handlungsaufforderung und einen QR-Code. Dieser Code verweist auf die mobile Teilnehmendenansicht des aktuell gestarteten Durchlaufs. Zusätzlich werden ein Beitrittscode und die direkte Teilnahme-URL angezeigt. Damit besteht neben dem QR-Code eine zweite Einstiegsmöglichkeit, falls ein QR-Code auf dem Videowürfel aus Distanz nicht zuverlässig erkannt wird oder Teilnehmende den Code manuell eingeben müssen. Der Wartezustand der Präsentationsansicht zeigt QR-Code, Beitrittscode und Teilnahme-URL (siehe Abbildung 27).



Abbildung 27: Präsentationsansicht im Wartezustand eines gestarteten Durchlaufs
Quelle: Eigene Aufnahme

Während des Spiels wechselt die Präsentationsansicht abhängig vom Spielzustand zwischen Wartebildschirm, Frage, Auflösung und Abschluss. Die Darstellung unterscheidet dabei zwischen Quiz, Bilderquiz und Kreuzworträtsel. Der Videowürfel dient nicht zur Dateneingabe, sondern zur gemeinsamen Orientierung des Publikums. Interaktive Aktionen bleiben der mobilen Teilnehmendenansicht und dem Administrationsbereich vorbehalten.

9.1.4 Mobile Teilnehmendenansicht

Die mobile Teilnehmendenansicht ist auf die Teilnahme mit einem Smartphone ausgerichtet. Nach dem Öffnen des QR-Codes oder der Teilnahme-URL tritt das Gerät dem aktiven Durchlauf bei. Die Oberfläche zeigt die zugewiesene Teilnahmenummer, den aktuellen Spielzustand und die jeweils mögliche Aktion. Im Wartezustand informiert sie darüber, dass der Durchlauf vorbereitet ist. Während einer Frage stellt sie die Antwortmöglichkeiten oder das Eingabefeld des jeweiligen Spielmodus bereit. Ein Beispiel für den Wartezustand auf dem Smartphone zeigt die reduzierte mobile Darstellung (siehe Abbildung 28).



Abbildung 28: mobile Teilnehmendenansicht im Wartezustand
Quelle: Eigene Aufnahme

Die mobile Teilnehmendenansicht vermeidet zusätzliche Navigation. Der Ablauf folgt dem Spielzustand und führt die Teilnehmenden schrittweise durch Beitritt, Antwortabgabe, Auflösung und Ergebnis. Diese Gestaltung unterstützt die Nutzung im Stadion, da nur kurze Interaktionen erforderlich sind. Die Teilnahmenummer und der Punktestand bleiben sichtbar, damit der eigene Status nachvollziehbar bleibt. In der Ergebnisansicht wird explizit zwischen Gewinn und Nicht-Gewinn unterschie-

den. Gewinnende Personen erhalten den Hinweis zur Abholung des Gewinns, während alle anderen Teilnehmenden eine explizite Abschlussmeldung erhalten.

Beim Kreuzworträtsel wurden die Eingabe und Auflösung auf die mobile Nutzung ausgerichtet. Felder können direkt fokussiert werden, auch wenn zuvor noch kein Wort aktiv ausgewählt wurde. Kreuzende Buchstaben werden zwischen den betroffenen Wörtern synchron geführt, damit ein eingegebener Buchstabe nicht mehrfach gepflegt werden muss. Sobald ein Wort vollständig ausgefüllt ist, kann es automatisch abgegeben werden. In der Auflösung werden die Buchstaben statisch dargestellt, sodass keine Eingabefelder mehr fokussiert oder Texte versehentlich markiert werden.

9.1.5 Responsives Verhalten und Zustandsführung

Die drei Oberflächen besitzen unterschiedliche Anforderungen an das Layout. Der Administrationsbereich ist für normale Bildschirme optimiert, weil dort Tabellen, Formulare und Verwaltungsaktionen parallel sichtbar sein müssen. Die Präsentationsansicht verwendet den verfügbaren Bildschirmraum (794*544 Pixel) von dem Videowürfel der Lakers vollständig und priorisiert Lesbarkeit aus Distanz. Die mobile Teilnehmendenansicht richtet Inhalte vertikal aus und beschränkt die Bedienung auf wenige zentrale Elemente.

Der sichtbare Zustand der Bedienoberfläche wird aus den Daten des Backends abgeleitet. Listenansichten laden gespeicherte Spielvorlagen und Durchläufe. Die Spielansichten reagieren auf den jeweils aktuellen Durchlaufzustand. Dadurch entsteht kein eigener Ablauf im Frontend. Die Oberfläche zeigt den Zustand an und löst erlaubte Aktionen aus, während die Entscheidung im Backend verbleibt.

9.2 Herausforderungen und Lösungsansätze

Während der Umsetzung traten verschiedene technische Herausforderungen auf, die sich vor allem aus dem Live-Charakter der Anwendung ergaben. Die Anwendung besteht nicht nur aus einer klassischen Weboberfläche mit einzelnen Formularen, sondern aus mehreren gleichzeitig aktiven Ansichten. Administrationsbereich, Präsentationsansicht und mobile Teilnehmendenansicht müssen denselben Spielzustand verwenden und während eines laufenden Spiels synchron bleiben. Zusätzlich mussten die drei Spielmodi so umgesetzt werden, dass sie gemeinsame Abläufe teilen, aber trotzdem ihre unterschiedlichen Eingabe- und Darstellungsformen unterstützen.

9.2.1 Synchronisation des Spielzustands

Eine zentrale Herausforderung war die Synchronisation zwischen Präsentationsansicht, Mobilgeräten und Administrationsbereich. Während eines Spiels ändern sich Zustand, Timer, aktuelle Frage, Antwortphase und Resultate laufend. Eine reine REST-basierte Kommunikation hätte dazu geführt, dass die Clients den Zustand regelmässig abfragen müssten. Das wäre nicht notwendiger Overhead und würde den Teilnehmenden und unserer Infrastruktur zusätzliche Last erzeugen.

Als Lösungsansatz wurde eine WebSocket-basierte Kommunikation über Socket.IO umgesetzt. Clients treten einem Raum pro Spiel-Durchlauf bei und erhalten bei jeder relevanten Änderung einen vollständigen Snapshot des aktuellen Spielzustands. Dadurch müssen die Clients nicht selbst rekonstruieren, welche Einzelereignisse zuvor passiert sind. Wenn ein Client später beitritt oder die Verbindung neu aufgebaut wird, kann er direkt den aktuellen Zustand erhalten und korrekt weiterlaufen.

9.2.2 Verwaltung von Live-Zustand und Persistenz

Nicht alle Daten eines laufenden Spiels eignen sich gleichermassen für die persistente Speicherung. Dauerhafte Informationen wie Durchlauf, Teilnehmende, Antworten, Punkte und Gewinndaten werden in der Datenbank gespeichert. Kurzlebige Informationen wie Timer, aktuelle Frage, Pausenstatus oder der nächste automatische Zustandswechsel sind dagegen Teil des Live-Zustands.

Das Backend speichert deshalb nur den persistenzrelevanten Teil in PostgreSQL, verwaltet den aktiven Spielzustand aber während des Durchlaufs im Arbeitsspeicher. Dadurch lassen sich Timer und automatische Übergänge einfach steuern, ohne jedes interne Live-Ereignis als Datenbankeintrag modellieren zu müssen. Gleichzeitig bleiben die für Auswertung und Nachvollziehbarkeit relevanten Daten erhalten.

9.2.3 Gemeinsame Struktur für unterschiedliche Spielmodi

Die drei Spielmodi Quiz, Kreuzworträtsel und Bilderrätsel unterscheiden sich fachlich deutlich. Quiz und Bilderrätsel arbeiten mit auswählbaren Antwortoptionen, während das Kreuzworträtsel Texteingaben verwendet. Beim Bilderrätsel kommen zusätzlich Bildreferenzen und Filterdaten hinzu. Trotzdem sollten nicht drei vollständig getrennte Daten- und Ablaufmodelle entstehen.

Gelöst wurde dies durch ein gemeinsames Grundmodell für Spiele, Fragen und Antworten. Der Spielmodus bestimmt, wie die Daten interpretiert und in der Oberfläche dargestellt werden. Gemeinsame Felder wie Frage, Reihenfolge, Punkte und Zeitlimit werden wiederverwendet. Spezifische Informationen wie Filterdaten oder spielmodusabhängige Einstellungen werden über flexible JSONB-Felder gespeichert. Dadurch bleibt das Datenmodell überschaubar und kann später um weitere Spielmodi erweitert werden.

9.2.4 Steuerung während eines Live-Spiels

Ein Live-Spiel im Stadion muss nicht nur automatisch ablaufen, sondern auch durch Administrierende steuerbar bleiben. Während der Umsetzung zeigte sich, dass feste Timer allein nicht ausreichen. Es muss möglich sein, Fragen manuell weiterzuschalten, ein Spiel zu pausieren oder einen Durchlauf vorzeitig zu beenden.

Der Spielablauf wurde deshalb als Zustandsmodell umgesetzt. Ein Durchlauf wechselt zwischen Lobby, Frage, Antwortanzeige und Abschluss. Automatische Timer lösen Zustandswechsel aus, können aber durch Admin-Aktionen übersteuert werden. Beim Pausieren wird die verbleibende Zeit gespeichert und beim Fortsetzen wiederverwendet. Dadurch bleibt der Ablauf planbar, ohne die Kontrolle während einer realen Durchführung zu verlieren.

9.2.5 Deployment und Konfiguration

Eine weitere Herausforderung war das Deployment der Anwendung ausserhalb der lokalen Entwicklungsumgebung. Frontend, Backend, Datenbank, Storage und Konfiguration müssen zusammen funktionieren. Zusätzlich dürfen sensible Werte wie Datenbankpasswörter oder JWT-Secrets nicht fest im Code hinterlegt werden.

Als Lösungsansatz wurde die Anwendung containerisiert und über eine Cloud-Umgebung bereitgestellt. Das Frontend und Backend laufen als getrennte Container, während die Datenbank über PostgreSQL bereitgestellt wird. Konfigurationswerte werden über Umgebungsvariablen und Secret-Mechanismen gesetzt. Migrationen sorgen dafür, dass das Datenbankschema reproduzierbar aufgebaut werden kann. Dadurch ist die Anwendung nicht nur lokal ausführbar, sondern auch in einer produktiven Umgebung betreibbar.

9.3 Deployment und Ausführung

Für den Betrieb der Anwendung wurde im Rahmen der Bachelorarbeit eine Architektur auf Basis von AWS umgesetzt. Dieses Kapitel dokumentiert die Evaluation der Betriebsoptionen, die daraus abgeleitete Entscheidung sowie die technische Umsetzung des Deployments.

9.3.1 Evaluation der Betriebsoptionen

Für den Betrieb der Anwendung wurden verschiedene Infrastrukturansätze evaluiert. Dabei wurden insbesondere die Kriterien Wartbarkeit, Skalierbarkeit, Betriebsaufwand, Unterstützung moderner Deployment-Prozesse, Datenschutz, Kosten sowie die technischen Anforderungen der Anwendung berücksichtigt.

Die Anwendung benötigt neben statischen Inhalten ein Backend für Spielzustände, Administrationsfunktionen, WebSocket-Kommunikation, Datei-Uploads und Datenbankzugriffe. Das Deployment soll zudem automatisiert über die GitLab-Pipeline für kontinuierliche Integration und kontinuierliches Deployment (CI/CD) erfolgen. Das Pausenspiel wird nur während ausgewählter Heimspiele, Tests oder Vorführungen verwendet. Deshalb war relevant, ob laufende Ressourcen ausserhalb der Nutzungszeiten reduziert werden können.

Kriterium	**Virtueller privater Server**	**Statisches Deployment**	**AWS**
Containerbetrieb	Möglich	Nicht geeignet	Voll unterstützt
CI/CD-Integration	Teilweise manuell	Eingeschränkt	Gut integrierbar
Backend und WebSockets	Möglich	Nicht geeignet	Voll unterstützt
Datenbankbetrieb	Selbst verwaltet	Nicht abgedeckt	Verwalteter Dienst möglich
Bildspeicherung	Selbst verwaltet	Teilweise möglich	Über S3 abgedeckt
Geheimnisverwaltung	Selbst umzusetzen	Eingeschränkt	Zentral über SSM
Protokollierung und Überwachung	Selbst aufzubauen	Eingeschränkt	Über CloudWatch möglich
Skalierbarkeit	Begrenzt	Begrenzt	Flexibel
Betriebsaufwand	Hoch	Niedrig	Mittel
Kostensteuerung	Manuell	Niedrige Kosten, aber funktional nicht geeignet	Gezieltes Stoppen von Rechenressourcen möglich

Tabelle 41: Vergleich der evaluierten Betriebsoptionen

Ein klassischer virtueller Server wäre geeignet gewesen, um Frontend, Backend und Datenbank zu betreiben. Diese Variante hätte jedoch einen höheren administrativen Aufwand verursacht. Insbesondere Systemwartung, Sicherheitsupdates, Backup-Management, Monitoring, Skalierung und Wiederherstellung müssten weitgehend selbst umgesetzt und betrieben werden.

Eine ausschliesslich statische Deployment-Lösung wurde ausgeschlossen, da sie die Anforderungen der Anwendung nicht erfüllt. Das Frontend könnte zwar statisch ausgeliefert werden, das System benötigt jedoch zusätzlich serverseitige Logik, WebSocket-Kommunikation, Datenbankzugriffe, Datei-Uploads und administrative Funktionen.

AWS erfüllt die Anforderungen des Projekts insbesondere hinsichtlich Skalierbarkeit, Wartbarkeit, Betriebssicherheit und automatisiertem Deployment. Die Plattform stellt dafür kombinierbare Dienste bereit, die zu den technischen und betrieblichen Anforderungen des Systems passen.

9.3.2 Entscheidung für AWS

Aufgrund der Evaluation fiel die Entscheidung auf AWS. Die Lösung unterstützt die containerisierte Architektur der Anwendung und lässt sich in die bestehende GitLab-CI/CD-Pipeline integrieren. Dadurch können die Erstellung, die Veröffentlichung der Containerimages und das Deployment automatisiert durchgeführt werden.

Durch die Nutzung von AWS-Diensten können Persistenz, Bildspeicherung, Geheimnisverwaltung und Protokollierung zentral verwaltet werden. Gleichzeitig erlaubt die Infrastruktur eine flexible Skalierung sowie das gezielte Herunterfahren laufender Container ausserhalb der Nutzungszeiten.

Ein weiterer Vorteil ergibt sich durch die Wahl der Region `eu-central-2` (Zürich). Die Datenhaltung in der Schweiz beziehungsweise im europäischen Raum unterstützt die Datenschutzanforderungen des Projekts und ermöglicht geringe Latenzzeiten für den vorgesehenen Einsatzort.

9.3.3 Deployment-Pipeline und Erstellungsprozess

Das Deployment erfolgt über die GitLab-CI/CD-Pipeline. Diese besteht aus den Phasen `validate`, `test`, `build`, `publish` und `deploy`. Für Änderungen an Frontend, Backend, Skripten oder der Pipeline führt sie Linting, Tests und Buildjobs aus. Bei Deployments veröffentlicht sie zusätzlich die Containerimages.

Der Erstellungsprozess ist in zwei Ebenen unterteilt. Zuerst prüfen die Buildjobs, ob Frontend und Backend unabhängig voneinander gebaut werden können. Anschliessend erzeugen die Publishjobs die Containerimages und veröffentlichen sie in ECR. Die Images werden mit dem Commit-Hash und zusätzlich mit der Kennzeichnung `latest` versehen. Dadurch kann eine konkrete Pipeline-Ausführung eindeutig einem Abbild zugeordnet werden.

Das Frontend-Abbild wird zweistufig gebaut. Im ersten Schritt erzeugt Node.js den Vite-Build. Im zweiten Schritt wird nur das erzeugte `dist`-Verzeichnis in ein Nginx-Abbild übernommen. Das Backend-Abbild wird ebenfalls zweistufig erstellt. Dabei wird zuerst die NestJS-Anwendung kompiliert. Das Laufzeitabbild enthält danach nur die notwendigen npm-Abhängigkeiten, die kompilierten Dateien und die für RDS benötigten Zertifikatsdateien.

Im Deployment-Schritt rendert die Pipeline aus den Abbild-URLs und den konfigurierten Umgebungsvariablen eine ECS-Task-Definition. Diese Task-Definition wird als neue Revision registriert und anschliessend für den ECS-Fargate-Service verwendet. Die Pipeline enthält zusätzlich manuelle Betriebsaufgaben, mit denen die Anwendung gestartet, gestoppt oder die Datenbank zurückgesetzt werden kann. Der Ablauf der Pipeline ist dargestellt (siehe Abbildung 29).

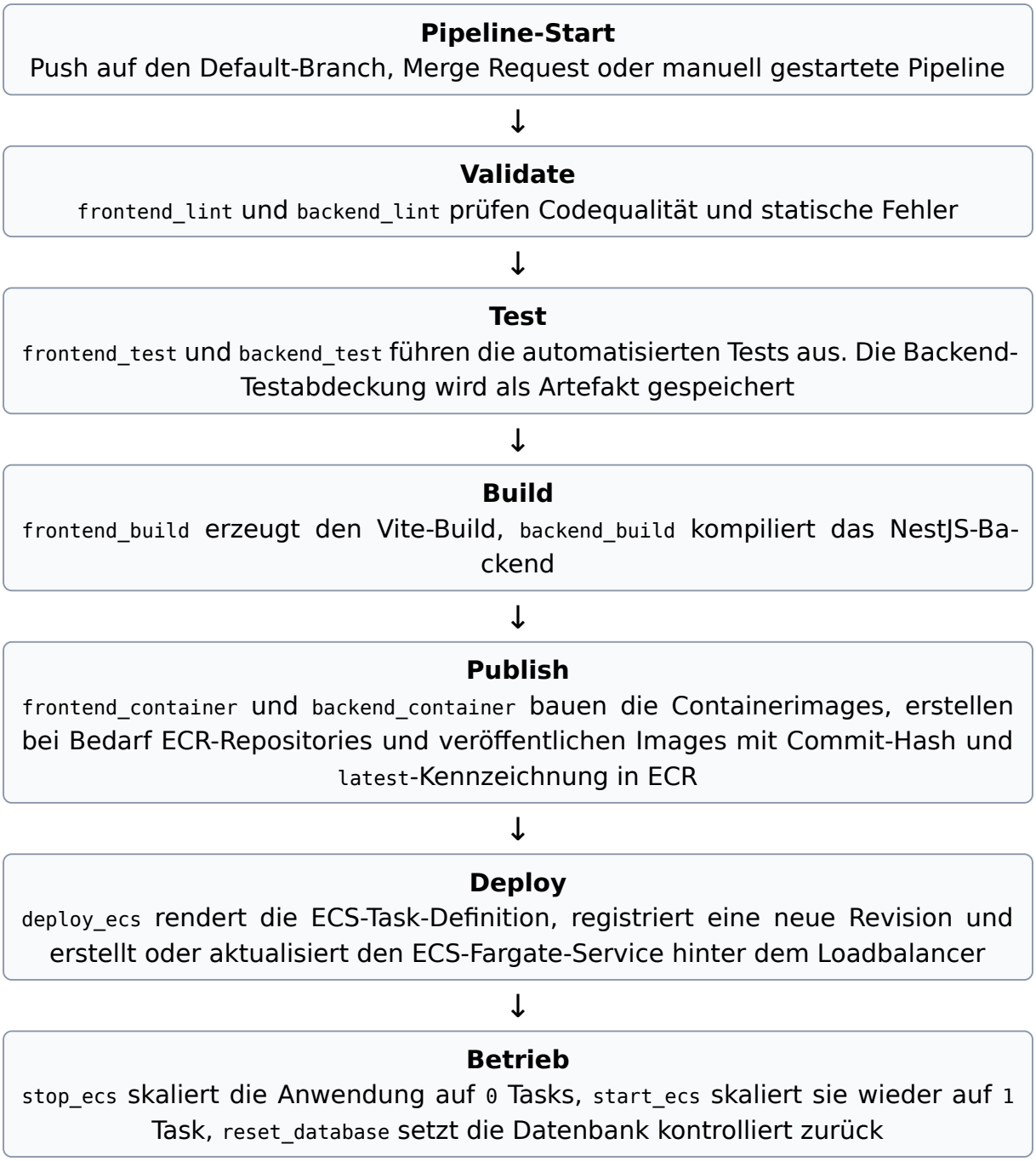


Abbildung 29: Ablauf der GitLab-CI/CD-Pipeline für Erstellung, Deployment und Kostensteuerung
Quelle: Eigene Abbildung

9.3.4 AWS Architektur

Die Laufzeitumgebung verwendet ECS mit Fargate für den Containerbetrieb. ECR speichert die Containerimages. Aurora PostgreSQL, S3, SSM, CloudWatch Logs, CloudFront und ein Loadbalancer ergänzen die Infrastruktur. Die Anwendungskomponenten werden containerisiert betrieben. Frontend und Backend werden als separate Images aus ECR bezogen.

Im Betrieb laufen Frontend- und Backend-Container gemeinsam in einer ECS-Fargate-Task. Der Frontend-Container liefert den Vite-Build über Nginx aus und behandelt clientseitiges Routing über einen Fallback auf `index.html`. Anfragen an `/api` sowie `/socket.io` werden von Nginx an das Backend weitergeleitet. Für WebSockets setzt die Nginx-Konfiguration die erforderlichen Upgrade-Header. Der Loadbalancer leitet öffentlichen HTTP-Verkehr an den Frontend-Container auf Port `80` weiter. CloudFront dient als HTTPS-Einstiegspunkt vor dem Loadbalancer. Innerhalb der Task kommuniziert Nginx mit dem Backend über `127.0.0.1:3000`. Dadurch sind API- und WebSocket-Anfragen für den Browser unter derselben öffentlichen Adresse erreichbar. Eine zusätzliche interne Diensterkennung zwischen Frontend und Backend wird nicht benötigt.

Die ECS-Task-Definition beschreibt beide Container, die Portzuordnungen, Zustandsprüfungen, Protokollkonfigurationen sowie die benötigten Umgebungsvariablen. Die Zustandsprüfung des Backends prüft den Endpunkt `/api/health`. Die Zustandsprüfung des Frontends kontrolliert die Auslieferung über Nginx. Die Containerprotokolle werden nach CloudWatch Logs geschrieben.

Die Datenbank wird als Aurora PostgreSQL betrieben. Das Backend verbindet sich über SSL mit der Datenbank und führt TypeORM-Migrationen beim Start aus, wenn dies über die Deployment-Konfiguration aktiviert ist. Secrets wie das Datenbankpasswort und das JWT-Secret werden nicht im Repository gespeichert, sondern über SSM in die ECS-Task eingebunden. Für den Zugriff auf AWS-Dienste werden IAM-Rollen verwendet. Die ECS-Ausführungsrolle erlaubt unter anderem das Laden der Containerimages, das Schreiben von Protokollen und das Lesen der SSM-Parameter. Die ECS-Task-Rolle stellt der laufenden Anwendung Berechtigungen für S3 und weitere benötigte AWS-Dienste bereit. Hochgeladene Bilder und Sponsorlogos werden in S3 gespeichert. Die vereinfachte AWS-Architektur ist dargestellt (siehe Abbildung 30).

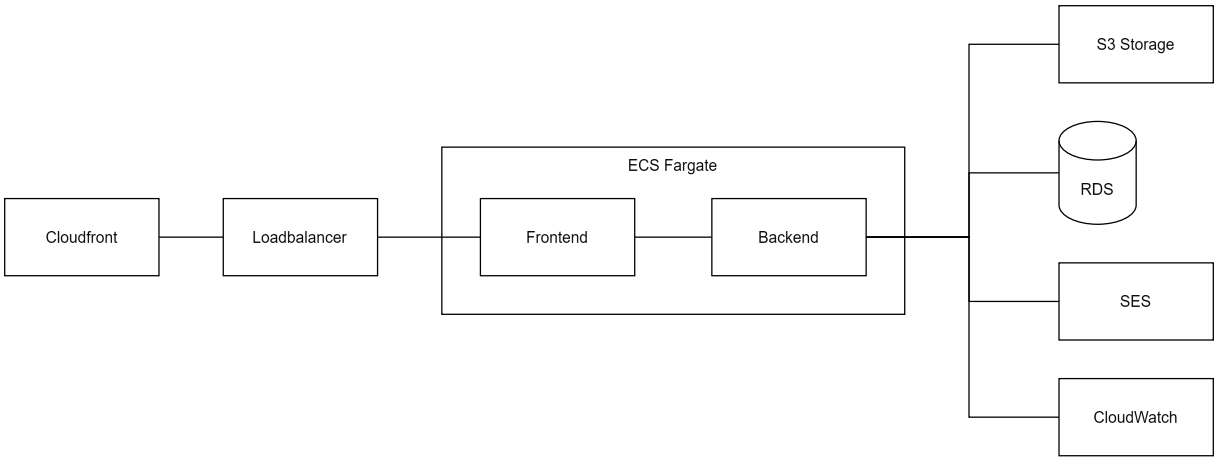


Abbildung 30: AWS-Deployment-Architektur
Quelle: Eigene Abbildung, erstellt mit draw.io

9.3.5 Kosten und Betrieb

Ein zentraler Betriebsaspekt sind die laufenden Kosten. Das Pausenspiel wird nicht dauerhaft benötigt, sondern kommt bei Heimspielen, Tests oder Vorführungen zum Einsatz. Die Architektur ist deshalb darauf ausgelegt, laufende Rechenressourcen ausserhalb der Nutzungszeiten zu reduzieren.

Die Anwendung auf ECS Fargate kann über die in Kapitel 9.3.3 beschriebenen Betriebsaufgaben auf 0 Tasks skaliert werden. Damit fallen für Frontend und Backend keine laufenden Fargate-Rechenkosten an, solange die Anwendung nicht verwendet wird. Vor einem Einsatz kann sie über denselben Ablauf wieder gestartet und auf den aktuellen Stand bereitgestellt werden.

Ein Teil der Kosten bleibt bestehen. Gespeicherte Daten, S3-Objekte, Container-images, CloudWatch Logs und der Loadbalancer bleiben als laufende Kostenfaktoren bestehen. Diese Aufteilung ist geeignet, weil der am stärksten nutzungsabhängige Teil der Anwendung gezielt deaktiviert werden kann.

9.3.6 Risiken

Wenn die Anwendung auf 0 Tasks skaliert ist, ist sie nicht erreichbar. Vor einem Einsatz muss sie deshalb rechtzeitig gestartet und getestet werden. Ein weiteres Risiko ist die höhere Komplexität von AWS. Fehlkonfigurationen können Kosten oder Sicherheitsprobleme verursachen, weshalb nicht benötigte Images, Protokolle und Testdaten regelmässig bereinigt werden sollten.

9.4 Testen und Qualitätssicherung

Die Qualitätssicherung der Anwendung kombiniert automatisierte Tests, statische Codeprüfungen, Laufzeitprüfungen und manuelle Funktionstests. Diese Massnahmen reduzieren Risiken, die im Stadionbetrieb unmittelbar sichtbar würden. Zu diesen Risiken gehören unberechtigter Zugriff auf den Administrationsbereich, fehlerhafte Spielzustände, ungültige Antworten, instabile Spielrunden und Probleme beim Beitritt der Teilnehmenden.

9.4.1 Backend-Tests

Im Backend werden die Tests mit `Jest`, `ts-jest` und `Supertest` ausgeführt. Der Schwerpunkt liegt auf Services, Controllern, Guards, Konfiguration, Spielablauf, Antwortlogik und Dateiablage. Externe Abhängigkeiten wie Repositorys oder AWS-Dienste werden in den Tests durch Testdoubles ersetzt. Dadurch kann die Fachlogik unabhängig von einer realen Datenbank oder Cloud-Infrastruktur geprüft werden. Ein grundlegender End-to-End-Test prüft zusätzlich das Startverhalten der NestJS-Anwendung.

9.4.2 Frontend-Tests

Im Frontend werden `Vitest`, Testing Library und `jsdom` eingesetzt. Die Tests prüfen Komponenten, Hooks, den API-Client, den Authentifizierungsablauf, Tabellen, Formulare, Pagination, Bild-Uploads und Hilfsfunktionen für die verschiedenen Spielmodi. Die Tests simulieren Eingaben der Benutzenden und prüfen Lade-, Fehler- und Erfolgszustände. Damit werden vor allem die wiederverwendbaren Bausteine der Bedienoberfläche abgesichert.

9.4.3 Statische Codeprüfung

Die statische Qualitätssicherung erfolgt mit ESLint und Prettier. Im Backend nutzt ESLint eine typgeprüfte TypeScript-Konfiguration. Prettier ist dort direkt in ESLint eingebunden.

Im Frontend prüft ESLint drei zentrale Bereiche: allgemeine JavaScript-Regeln, React Hooks und React Refresh. Dafür werden die empfohlenen JavaScript-Regeln von ESLint, das empfohlene Regelset für React Hooks und die Vite-Konfiguration von React Refresh verwendet (Barre, o. J.; Meta Platforms, Inc., o. J.; OpenJS Foundation and ESLint contributors, o. J.). Diese Auswahl entspricht zentralen Bestandteilen der offiziellen Vite-React-TypeScript-Vorlage (Vite contributors, o. J.). Die vollständige TypeScript-ESLint-Recommended-Konfiguration aus dieser Vorlage wird im Projekt jedoch nicht übernommen. Für TypeScript-Dateien wird stattdessen der TypeScript-Parser verwendet und die Regel `@typescript-eslint/no-unused-vars` aktiviert.

Damit erkennt ESLint unter anderem ungenutzte Variablen, fehlerhafte Hook-Verwendungen und Komponentenexporte, die nicht mit React Refresh kompatibel sind.

9.4.4 CI/CD und Testabdeckung

Die CI/CD-Pipeline führt Linting, Tests und die Erstellung der Anwendung automatisiert aus. Der Backend-Testjob erzeugt zusätzlich einen Bericht zur Testabdeckung. Die dort ausgewiesene Backend-Testabdeckung liegt bei 70.26%.

Diese Kennzahl zeigt, dass ein Grossteil der Backend-Logik durch automatisierte Tests geprüft wird. Sie ersetzt jedoch keine fachliche Bewertung der geprüften Risiken. Deshalb wird die Testabdeckung durch manuelle Tests und die in Kapitel 10.3 beschriebenen API-Lasttests ergänzt. Die zusammenfassende Validierung der funktionalen und nichtfunktionalen Anforderungen erfolgt in Kapitel 10.2.1 und Kapitel 10.2.2.

9.4.5 Laufzeitprüfungen

Ein Teil der Qualitätssicherung findet zur Laufzeit statt. Das Backend validiert Umgebungsvariablen beim Start mit Joi. Fehlende oder ungültige Konfigurationswerte werden dadurch früh erkannt. Eingehende API-Daten werden über Data Transfer Objects (DTOs) und die globale NestJS-`ValidationPipe` geprüft. Für numerische Pfadparameter werden `ParseIntPipes` verwendet. Zusätzlich stellt das Backend den Endpunkt `/api/health` bereit. Dieser Healthcheck prüft neben der API auch die Verbindung zur Datenbank. Der Backend-Container startet erst nach einer erfolgreichen Datenbankprüfung.

9.4.6 Manuelle Funktionstests

Automatisierte Tests decken nicht alle Qualitätsmerkmale ab. Die Lesbarkeit auf dem Videowürfel, die Bedienbarkeit auf Smartphones und der Ablauf einer vollständigen Spielrunde wurden deshalb manuell geprüft. Während der Umsetzung wurden Spielinhalte im Administrationsbereich erfasst, Spiele gestartet, Teilnehmende über den mobilen Link angemeldet, Antworten abgegeben und Resultate auf dem Grossbildschirm kontrolliert.

9.5 Zusammenfassung

Die Umsetzung führte zu einem funktionsfähigen MVP für ein interaktives Pausenspiel. Die Anwendung deckt den vollständigen Ablauf von der Vorbereitung der Spielinhalte bis zur Auswertung eines beendeten Durchlaufs ab. Zum umgesetzten Umfang gehören der Administrationsbereich, die mobile Teilnehmendenansicht, die Präsentationsansicht für den Videowürfel, die drei Spielmodi Quiz, Bilderquiz und Kreuzworträtsel sowie die serverseitige Steuerung des Spielzustands.

9.5.1 Umsetzungskern

Der Administrationsbereich bildet die Inhaltspflege und die operative Spielsteuerung ab. Spielvorlagen, Fragen, Antworten und Sponsoren können ohne Anpassung des Quellcodes verwaltet werden. Beim Start einer Spielvorlage erzeugt das Backend einen Durchlauf und führt diesen über ein Zustandsmodell mit Lobby, Fragephase, Auflösung und Abschluss. Die fachliche Logik bleibt dabei im Backend. Das Frontend zeigt den aktuellen Zustand an und löst nur die dafür vorgesehenen Aktionen aus.

Die Synchronisation der Ansichten erfolgt über Socket.IO. Das Backend erzeugt dafür Zustandsobjekte, die den jeweils relevanten Ausschnitt des aktuellen Durchlaufs enthalten. Administrationsbereich, Präsentationsansicht und mobile Teilnehmendenansicht erhalten dieselben fachlichen Informationen und reagieren dadurch konsistent auf Phasenwechsel, Pausen, Antwortabgaben und den Spielabschluss. Die Antwortverarbeitung, Punkteberechnung, Rangbildung und Gewinnabwicklung erfolgen ebenfalls serverseitig.

9.5.2 Technische Grundlage

Die technische Grundlage umfasst eine persistente Datenhaltung mit PostgreSQL, eine strukturierte Backend-Architektur mit NestJS, eine React-basierte Einseitenanwendung sowie die Ablage von Medien in S3-kompatiblem Speicher. Administrationsfunktionen sind durch eine Anmeldung geschützt. Die öffentlichen Ansichten für Teilnehmende und Videowürfel bleiben davon getrennt. Für Entwicklung und Betrieb wurden Containerisierung, Umgebungsvariablen, Deployment-Abläufe und automatisierte Tests eingerichtet.

9.5.3 Einordnung und Ausblick

Einzelne Anforderungen bleiben als Erweiterung offen. Die direkte Integration in die Lakers-App wurde nicht umgesetzt, die Identifikation über Sitzungstoken lässt eine spätere Anbindung jedoch zu. Ein saisonübergreifendes Scoreboard und ein fein abgestuftes Rollenmodell für unterschiedliche Administrationsrollen sind ebenfalls nicht Bestandteil des aktuellen MVP.

Die realisierte Lösung erfüllt damit den Zweck eines technisch belastbaren MVP. Sie zeigt, wie Stadionpublikum, Videowürfel und Spielleitung in einem gemeinsamen Ablauf verbunden werden können. Auf dieser Grundlage können weitere Spielmodi, zusätzliche Auswertungen, eine App-Integration sowie ergänzende Sicherheits- und Betriebsfunktionen gezielt ausgebaut werden.

10 Evaluation

Dieses Kapitel fasst die Evaluation des entwickelten MVP zusammen.

10.1 Evaluationsziel und Vorgehen

Ziel der Evaluation ist es, zu beurteilen, ob die entwickelte Anwendung als MVP für ein interaktives Pausenspiel der SCRJ Lakers geeignet ist. Dafür wird nicht nur die technische Funktionsfähigkeit geprüft, sondern auch, ob die Umsetzung die fachlichen Ziele und Qualitätsanforderungen aus Analyse, Requirements und Konzept erfüllt. Betrachtet werden die funktionalen und nichtfunktionalen Anforderungen, das Verhalten unter Last, die Bedienbarkeit für Zuschauende, die Qualitätssicherung, der Betrieb sowie Sicherheits- und Datenschutzaspekte. Damit wird sichtbar, welche Teile des Systems bereits für einen produktionsnahen Einsatz geeignet sind und welche Einschränkungen für eine Weiterentwicklung bestehen bleiben.

Als Bewertungsmethoden wurden Funktionstests, automatisierte Tests, manuelle Tests, Code-Analyse und API-Lasttests verwendet. Funktionale Anforderungen wurden durch das Durchspielen der zentralen Abläufe geprüft. Dazu gehören das Erfassen von Spielinhalten im Administrationsbereich, das Starten und Steuern einer Spielrunde, der Beitritt von Teilnehmenden, das Abgeben von Antworten und die Auswertung nach Spielende. Nichtfunktionale Anforderungen wurden je nach Kriterium durch Tests, Messungen oder Analyse der Architektur bewertet.

Die Bewertung unterscheidet zwischen erfüllt, teilweise erfüllt und nicht erfüllt. Eine Anforderung gilt als erfüllt, wenn die beschriebene Funktion oder Qualität im System umgesetzt und durch eine geeignete Verifikation plausibel nachgewiesen werden kann. Teilweise erfüllt bedeutet, dass der Kern der Anforderung umgesetzt ist, einzelne Aspekte jedoch fehlen oder nicht vollständig verifiziert werden konnten. Nicht erfüllt bedeutet, dass die Anforderung in der aktuellen Version nicht umgesetzt wurde oder das definierte Akzeptanzkriterium nicht erreicht.

10.2 Anforderungsvalidierung

10.2.1 Validierung der funktionalen Anforderungen

Dieses Unterkapitel bewertet die Umsetzung anhand der im Konzept definierten funktionalen Anforderungen. Alle funktionalen Anforderungen wurden durch Funktionstests überprüft.

ID	Kurztitel	Status	Begründung
FR011	App Teilnahme	Teilweise erfüllt	Die Teilnahme ist über den mobilen Link einer aktiven Spielrunde möglich. Die direkte Integration in die SCRJ Lakers-App wurde aufgrund sicherheitsrelevanten Anforderungen des Herstellers nicht umgesetzt.
FR021	Fragen beantworten	Erfüllt	Zuschauende können Quiz- und Kreuzworträtselantworten abgeben. Das Backend speichert und bewertet diese Antworten.
FR031	Lösung ansehen	Erfüllt	Die Auflösung einer Frage wird in der Präsentationsansicht angezeigt.
FR041	Spiel-Scoreboard ansehen	Teilweise erfüllt	Die persönliche Punktzahl und eine Rangliste im Administrationsbereich sind vorhanden. Ein öffentliches Scoreboard für alle Zuschauenden wurde nicht vollständig umgesetzt.
FR042	Saison-Scoreboard ansehen	Nicht erfüllt	Ein saisonübergreifendes Scoreboard ist in der aktuellen Umsetzung nicht vorhanden. Dies wurde für den MVP bewusst entschieden, da dafür eine vollständige Verwaltung von Benutzenden nötig wäre. Zudem müssten sich Zuschauende zusätzlich zur Lakers-App an einem zweiten Ort anmelden.
FR051	Gewinn anzeigen	Erfüllt	Die Gewinnnummer wird nach Spielende angezeigt. Gewinnende Personen können Kontaktdaten zur Gewinnabholung erfassen.
FR111	Quizfragen erfassen	Erfüllt	Quizfragen können im Administrationsbereich erfasst und gespeichert werden.
FR112	Quizantworten erfassen	Erfüllt	Antworten können erfasst und als korrekt markiert werden.
FR013	Kreuzworträtsel erfassen	Erfüllt	Kreuzworträtsel können über den entsprechenden Spielmodus erfasst werden. Die ID wird gemäss Konzeptdokument übernommen.

Tabelle 42: Validierung der funktionalen Anforderungen

ID	Kurztitel	Status	Begründung
FR014	Bilder erfassen	Erfüllt	Bilder können für Bilderrätsel und Sponsoren hochgeladen werden.
FR015	Bildeffekt anpassen	Erfüllt	Bildeffekte werden als Filterdaten gespeichert und im Bilderrätsel angewendet.
FR121	Spielmodus konfigurieren	Erfüllt	Im Administrationsbereich kann zwischen verschiedenen Spielmodi gewählt werden.
FR122	Spieldauer anpassen	Erfüllt	Fragen können mit einer eigenen Zeitdauer gespeichert werden. Dadurch lässt sich die Rundendauer pro Frage beeinflussen.
FR131	Sponsor integrieren	Erfüllt	Sponsoren können erfasst und einem Spiel zugeordnet werden.
FR211	Spiel laden	Erfüllt	Vorbereitete Spiele können geladen und in der Spielübersicht angezeigt werden.
FR212	Spiel starten	Erfüllt	Ein Spiel kann aus dem Administrationsbereich gestartet werden. Das Backend erzeugt dabei eine neue Spielrunde.
FR213	Nächste Frage anzeigen	Erfüllt	Die Spielleitung kann während einer Spielrunde manuell zur nächsten Phase wechseln.
FR214	Spiel pausieren	Erfüllt	Eine laufende Spielrunde kann pausiert und wieder fortgesetzt werden.
FR221	Spiel beenden	Erfüllt	Eine laufende Spielrunde kann beendet werden.
FR222	Spiel auswerten	Erfüllt	Für abgeschlossene Spielrunden werden Statistiken, Antworten und Ranglisten im Administrationsbereich angezeigt.

Tabelle 43: Validierung der funktionalen Anforderungen

Von den 20 funktionalen Anforderungen sind 17 erfüllt, zwei teilweise erfüllt und einer nicht erfüllt. Offen bleiben vor allem Funktionen, die über den unmittelbaren Spielablauf hinausgehen. Dazu gehört insbesondere das saisonübergreifende Scoreboard. Die Teilnahme wurde technisch über einen mobilen Link umgesetzt. Damit ist der Spielbeitritt möglich, die ursprünglich beschriebene Integration in die Lakers-App bleibt jedoch eine spätere Erweiterung.

10.2.2 Validierung der nichtfunktionalen Anforderungen

Dieses Unterkapitel bewertet die Umsetzung anhand der im Konzept definierten nichtfunktionalen Anforderungen. Die Bewertung orientiert sich an den FURPS-Kategorien und den jeweiligen Akzeptanzkriterien.

ID	Status	Verifikation	Begründung
NFR001	Erfüllt	Automatisierte Tests und Funktionstest	Der Administrationsbereich ist durch JWT-basierte Authentifizierung geschützt. Öffentliche und geschützte Endpunkte werden im Backend unterschieden.
NFR002	Erfüllt	Funktionstest und automatisierte Tests	Zuschauende treten einer aktiven Spielrunde über deren Spielrunden-ID bei. Nicht vorhandene oder nicht aktive Runden werden vom Backend abgewiesen.
NFR003	Erfüllt	Funktionstest	Spielinhalte können schrittweise gespeichert werden. Fragen benötigen beim Speichern gültige Antwortdaten, unvollständige Spielentwürfe können jedoch gespeichert werden.
NFR004	Nicht erfüllt	Funktionstest	Das System besitzt einen Zugriffsschutz für den Administrationsbereich, aber kein ausgebautes Rollenmodell für Marketing und Spielleitung.
NFR101	Erfüllt	Usertest	Alle vier Testpersonen konnten der Anwendung ohne Hilfe beitreten. Damit wurde das Akzeptanzkriterium von mindestens 3 von 4 Testpersonen erfüllt.

Tabelle 44: Validierung der nichtfunktionalen Anforderungen

ID	Status	Verifikation	Begründung
NFR102	Erfüllt	Manueller Test	Die mobile Teilnehmendenansicht wurde auf unterschiedlichen Geräten von unterschiedlichen Herstellern geprüft.
NFR201	Erfüllt	Funktionstest	Ungültige Antworten und einzelne Fehlbedienungen werden im Backend abgefangen. Die Spielrunde bleibt für andere Teilnehmende nutzbar.
NFR202	Erfüllt	Funktionstest	Antworten und Resultate werden persistiert. Bei einem Seiten-Reload oder einer kurzen Unterbrechung von Seiten der Benutzenden kann der aktuelle Status wieder geladen werden.
NFR301	Erfüllt	Funktionstest	Der Einstieg in eine aktive Spielrunde funktioniert ohne spürbare Ladezeiten.
NFR302	Erfüllt	API-Lasttest	Die API erfüllt das Zeitkriterium im erwarteten Lastbereich mit minimalen AWS-Ressourcen. Mit verdoppelten AWS-Ressourcen bleibt auch das 95%-Perzentil bei 1000 gleichzeitigen Benutzenden deutlich unter zwei Sekunden.

Tabelle 45: Validierung der nichtfunktionalen Anforderungen

ID	Status	Verifikation	Begründung
NFR303	Erfüllt	API-Lasttest	Der Lasttest mit 500 gleichzeitigen Benutzenden blieb ohne kritische Fehler und erfüllt damit das definierte Akzeptanzkriterium.
NFR401	Erfüllt	Codeanalyse	Neue Spielinhalte und Spielmodi können über klar abgegrenzte Datenstrukturen und Services ergänzt werden. Dafür sind Codeanpassungen nötig, der bestehende Kern muss jedoch nicht grundlegend umgebaut werden.
NFR402	Erfüllt	Funktionstest	Spiele, Sponsoren und Laufzeiten können über den Administrationsbereich oder gespeicherte Konfigurationen gepflegt werden.
NFR403	Erfüllt	Codeanalyse	Die Identifikation über Sitzungstoken ist umgesetzt. Die Trennung zwischen Sitzungstoken, Benutzende und Teilnahme ermöglicht eine spätere Anbindung an eine API der Lakers-App.

Tabelle 46: Validierung der nichtfunktionalen Anforderungen

Von den 14 nichtfunktionalen Anforderungen sind 13 erfüllt und einer nicht erfüllt. Die für den umgesetzten MVP zentralen Aspekte Zugriffsschutz, Spielbeitritt, Antwortverarbeitung und Lastverhalten im erwarteten Nutzungsbereich sind umgesetzt. Offen bleibt insbesondere das Rollenmodell im Administrationsbereich. Weitere Einschränkungen bestehen bei Anforderungen, deren Akzeptanzkriterien quantitative Messungen verlangen. Diese Messungen wurden nur für die API-Lasttests vollständig durchgeführt.

10.3 API-Lasttest

Die API-Lasttests hatten das Ziel, das Backend und die PostgreSQL-Datenbank unter steigender Last zu überprüfen. Da bei diesen Tests direkt die API angesprochen wurde, wurde bewusst nicht die Performance des Frontends gemessen. Das Frontend wird im Betrieb als statische Anwendung über nginx ausgeliefert und hat auf die Antwortzeiten der getesteten API-Endpunkte keinen direkten Einfluss. Der Test fokussiert deshalb auf die Verarbeitung der Anfragen im Backend, die Datenbankzugriffe und das Verhalten des AWS-Deployments unter gleichzeitiger Nutzung.

Die Testdaten stammen aus dem AWS-Deployment. Für den ersten Test wurde die produktive Umgebung mit minimaler Ressourcenkonfiguration verwendet (Backend: AWS ECS Fargate mit 0.25 vCPU und 0.5 GB RAM, Datenbank: einzelne Aurora PostgreSQL-Instanz mit einer ACU). Backend und Datenbank wurden damit bewusst klein dimensioniert betrieben. Damit sollte geprüft werden, wie weit die gewählte Minimalarchitektur für den erwarteten Einsatz im Stadion ausreicht.

Anschliessend wurde derselbe Lasttest mit verdoppelten AWS-Ressourcen wiederholt. Dabei wurden die Ressourcen für Backend und Datenbank erhöht (Backend: AWS ECS Fargate mit 0.5 vCPU und 1 GB RAM, Datenbank: Aurora PostgreSQL mit zwei ACU). Dieser zweite Test zeigt, wie stark sich eine einfache Skalierung der bestehenden AWS-Architektur auf die Antwortzeiten auswirkt.

Getestet wurden drei Laststufen mit steigender Anzahl gleichzeitiger Benutzender. Jede simulierte Person ist dem Spiel beigetreten und hat mehrere Antworten abgegeben. Die Laststufen wurden bewusst im Bereich des in Tabelle 25 definierten guten Mitmachwerts sowie deutlich darüber gewählt. Die folgenden Tabellen und die Visualisierungen fassen die wichtigsten Messwerte zusammen (siehe Abbildung 31 & Abbildung 32).

Benutzende	Durchschnitt in ms	Median in ms	90% Perzentil in ms	95% Perzentil in ms
100	24	13	35	68
500	155	17	583	1033
1000	542	70	1988	2625

Tabelle 47: Zusammenfassung des API-Lasttests mit minimalen AWS-Ressourcen

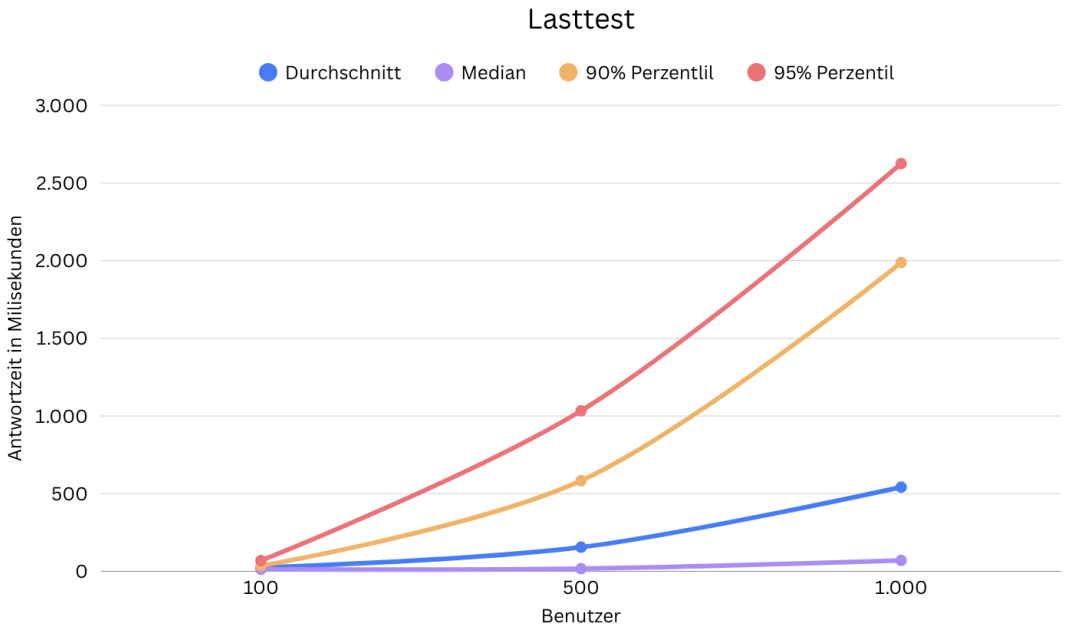


Abbildung 31: Visualisierung der API-Antwortzeiten mit minimalen AWS-Ressourcen
Quelle: Eigene Abbildung, erstellt mit Canva

Benutzende	Durchschnitt in ms	Median in ms	90% Perzentil in ms	95% Perzentil in ms
100	27	17	39	64
500	33	18	56	90
1000	47	20	112	223

Tabelle 48: Zusammenfassung des API-Lasttests mit verdoppelten AWS-Ressourcen

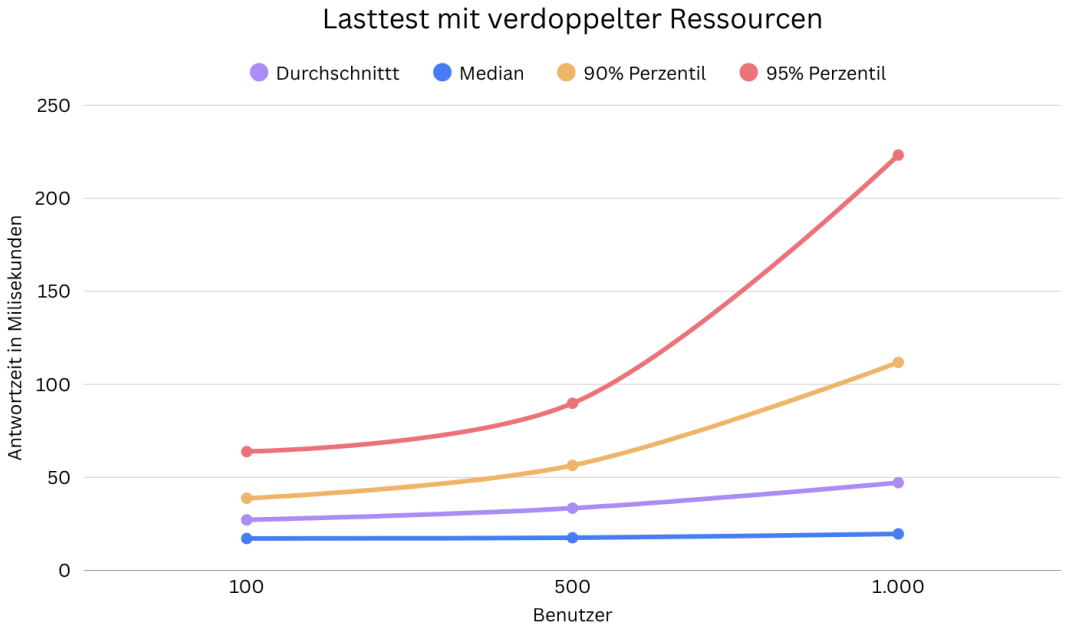


Abbildung 32: Visualisierung der API-Antwortzeiten mit verdoppelten AWS-Ressourcen
Quelle: Eigene Abbildung, erstellt mit Canva

Die Messergebnisse der Minimalressourcen zeigen, dass die API bei der niedrigsten Laststufe sehr kurze Antwortzeiten erreichte. Mit zunehmender Last steigen die Antwortzeiten jedoch deutlich an. Besonders die oberen Perzentile zeigen, dass einzelne Anfragen spürbar länger dauern können. Der Median bleibt zunächst tief, während die oberen Perzentile bereits deutlich höher liegen.

Bei der höchsten Laststufe werden die Antwortzeiten nochmals deutlich höher. Für eine interaktive Anwendung im Stadion ist dies ein Hinweis darauf, dass die Minimalressourcen bei hoher Last an ihre Grenze kommen.

Der zweite Lasttest mit verdoppelten AWS-Ressourcen zeigt ein deutlich stabileres Verhalten. Die Skalierung reduziert damit vor allem die Ausreisser bei hoher Last, während der Median in allen Laststufen weiterhin im niedrigen zweistelligen Millisekundenbereich bleibt.

Für den erwarteten produktiven Einsatz liegt dieses Ergebnis im definierten Zielbereich. Die Analyse in Tabelle 24 des Mitmachwerts zeigt, dass bereits eine vergleichsweise kleine aktive Teilnehmendengruppe einen guten Mitmachwert erreicht. Die niedrigste Laststufe liegt damit nahe an diesem Zielwert und blieb unter dem definierten Antwortzeitkriterium. Die höheren Laststufen liegen deutlich über dem erwarteten guten Mitmachwert und dienen deshalb vor allem als Belastungs- und Sicherheitsreserve. Die Messwerte mit verdoppelten Ressourcen zeigen, dass auch diese höheren Laststufen ohne grundlegende Architekturänderung mit tiefen Antwortzeiten bewältigt werden können.

Für den MVP kann deshalb weiterhin die kostenoptimierte Minimalvariante als Ausgangspunkt verwendet werden. Falls sich in späteren Einsätzen zeigt, dass deutlich mehr Personen gleichzeitig teilnehmen, können die AWS-Ressourcen erweitert werden. Der zweite Lasttest bestätigt, dass eine Erhöhung der ECS-Fargate-Ressourcen und der Datenbankkapazität ohne grundlegende Änderung der Anwendung wirksam ist. Für produktive Einsätze mit hoher Teilnahme sollte die Skalierung vor dem Spielbetrieb gezielt aktiviert und erneut überwacht werden.

10.4 Nutzungsbezogene Evaluation

10.4.1 Usability-Bewertung

Die Usability-Bewertung konzentriert sich auf den Einstieg in das Spiel, die Bedienbarkeit auf Smartphones und die Verständlichkeit des Spielablaufs. Diese Aspekte sind für den Einsatz im Stadion zentral, da Zuschauende das Spiel ohne vorgängige Schulung und unter zeitlichem Druck verwenden müssen.

Für die Bewertung wurde eine Umfrage mit vier Testpersonen durchgeführt, deren Ergebnisse im Anhang unter Umfrageergebnisse Usertest dokumentiert sind. Die Antworten wurden auf einer Skala von 1 bis 5 erhoben, wobei 5 die beste Bewertung darstellt. Ursprünglich war vorgesehen, den Endusertest direkt im Lakers-Stadion mit dem Videowürfel durchzuführen. Aufgrund von Bauarbeiten im Stadion war der Videowürfel jedoch vom Strom getrennt und konnte nicht verwendet werden. Der Test wurde deshalb auf einem Computerdisplay mit derselben Auflösung durchgeführt. Damit konnte die Darstellung der Präsentationsansicht technisch geprüft werden, während reale Stadionfaktoren wie Distanz, Lichtverhältnisse und Ablenkung nicht vollständig abgebildet werden konnten.

Aussage	Durchschnitt
Ich konnte der Anwendung ohne Hilfe beitreten.	5.00
Mir war klar, was ich während des Spiels tun muss.	4.75
Die Bedienung auf meinem Gerät war verständlich.	4.75
Die angezeigten Texte und Rückmeldungen waren verständlich.	4.25
Die Zeit pro Frage war ausreichend.	4.25
Ich würde das Spiel im Stadionkontext wieder spielen.	4.75

Tabelle 49: Zusammenfassung der Usability-Umfrage mit vier Testpersonen

10.4.2 Einstieg in die Spielrunde

Der Einstieg in eine aktive Spielrunde erfolgt über einen mobilen Link beziehungsweise über den vorgesehenen QR-Code. Als Reaktion auf das Feedback aus dem Usertest wird auf der Präsentationsansicht zusätzlich ein Beitrittscode mit Teilnahme-URL angezeigt. Dadurch bleibt der Einstieg auch dann möglich, wenn der QR-Code nicht direkt gescannt werden kann. Nach dem Öffnen der mobilen Teilnehmendenansicht können Teilnehmende einer laufenden Spielrunde beitreten und eine Antwort abgeben. Dieser Ablauf erfüllt den fachlichen Kern von NFR101, weil alle vier Testpersonen der Anwendung ohne Hilfe beitreten konnten. Das Akzeptanzkriterium fordert mindestens 3 von 4 Testpersonen. Da alle vier Testpersonen den Beitritt ohne Hilfe bestätigten, gilt NFR101 als erfüllt. Die ursprünglich geplante Integration in die Lakers-App wurde im MVP nicht umgesetzt. Dadurch bleibt der Einstieg technisch funktionsfähig, ist aber noch nicht vollständig in den Zielkanal der Lakers eingebettet.

10.4.3 Bedienbarkeit auf Smartphones

Die mobile Oberfläche wurde auf unterschiedlichen Bildschirmgrössen geprüft. Texte, Antwortoptionen und zentrale Bedienelemente bleiben ohne Zoomen bedienbar. Nach dem Usertest wurde die mobile Startansicht weiter reduziert, indem redundante Informationen entfernt wurden. Die Ergebnisansicht unterscheidet nun sichtbar zwischen Gewinn und Nicht-Gewinn, damit auch Teilnehmende ohne Gewinn eine eindeutige Rückmeldung erhalten. Für den MVP wird NFR102 deshalb teilweise erfüllt, da die Bedienbarkeit auf Smartphones geprüft und die Präsentationsansicht mit gleicher Auflösung getestet wurde, ein vollständiger Test unter echten Stadionbedingungen jedoch nicht durchgeführt werden konnte.

10.4.4 Verständlichkeit der Spielmodi

Die Spielmodi Quiz und Kreuzworträtsel sind für Zuschauende verständlich, weil sie bekannte Interaktionsmuster verwenden. Die Bedienoberfläche reduziert die notwendigen Aktionen auf den Beitritt zur Spielrunde und die Eingabe beziehungsweise Auswahl einer Antwort. Dadurch bleibt der Ablauf auch für Personen verständlich, die nur kurz während einer Drittelpause teilnehmen. In der Umfrage wurde das Bilderrätsel von drei der vier Testpersonen als verständlichster Modus genannt. Bei dem unterhaltend Spielmodus wurden Bilderrätsel und Kreuzworträtsel jeweils zweimal genannt.

10.4.5 Rückmeldungen aus dem Usertest

Die offenen Rückmeldungen zeigen mehrere konkrete Verbesserungspunkte. Mehrfach erwähnt wurde das Kreuzworträtsel, insbesondere die Eingabe auf mobilen Geräten und die mögliche Lesbarkeit auf kleinen Bildschirmen. Eine Testperson berichtete, dass unter iOS/Safari zwei Klicks nötig waren, bis die Tastatur geöffnet wurde. Weitere Rückmeldungen betrafen die Farbpalette, eine klarere Anzeige bei nicht gewonnenem Spiel, einen alternativen Beitrittscode zum QR-Code und die Markierbarkeit von Texten bei der Auflösung. Als unterstützende Aspekte wurden die geringe Anzahl notwendiger Aktionen, die Bedienung auf dem Testgerät und der Live-Charakter des Kreuzworträtsels genannt.

10.4.6 Umgesetzte Verbesserungen

Mehrere dieser Punkte wurden nach dem Usertest direkt umgesetzt. Die Präsentationsansicht zeigt neben dem QR-Code nun einen Beitrittscode und die Teilnahme-URL. Die Ergebnisansicht enthält eine explizite Nicht-Gewonnen-Meldung. Beim Kreuzworträtsel wurden Eingabe und Auflösung überarbeitet: Felder sind direkter fokussierbar, kreuzende Buchstaben werden synchronisiert, vollständig ausgefüllte Wörter können automatisch abgegeben werden und die Auflösung wird als statische Darstellung angezeigt. Dadurch wird die Bedienung in der Fragephase eindeutiger und die Markierbarkeit von Texten in der Auflösung vermieden.

10.4.7 Verbleibende Einschränkungen

Einschränkungen bestehen vor allem dort, wo der reale Stadionkontext nicht vollständig simuliert werden konnte. Dazu gehören Lichtverhältnisse, Ablenkung, Netzqualität, Geräuschpegel und der gleichzeitige Blickwechsel zwischen Smartphone und Videowürfel. Diese Faktoren sollten vor einem produktiven Einsatz in einem Stadiontest überprüft werden.

10.5 Gesamtbewertung und Limitationen

Die Evaluation zeigt, dass das entwickelte System die für den MVP definierten Kernabläufe abdeckt. Die zentralen Abläufe für Vorbereitung, Durchführung, Teilnahme und Auswertung einer Spielrunde sind umgesetzt.

Die funktionalen Anforderungen sind mehrheitlich erfüllt. Die offenen Punkte betreffen vor allem Funktionen ausserhalb des unmittelbaren Spielablaufs und sind in Kapitel 10.2.1 ausgewiesen. Für eine weiterentwickelte Produktversion bleiben diese Erweiterungen relevant.

Bei den nichtfunktionalen Anforderungen sind die wichtigsten MVP-Qualitäten ebenfalls erreicht. Die Detailbewertung in Kapitel 10.2.2 zeigt jedoch auch Einschränkungen beim Rollenmodell und bei der Stadionvalidierung. Bei sehr hoher Last zeigte der zweite API-Lasttest, dass verdoppelte AWS-Ressourcen die Antwortzeiten deutlich senken. Vor einem Einsatz mit grösserem Publikum sollte die gewählte Skalierung dennoch unter produktionsnahen Bedingungen überwacht werden.

Die grösste Limitation der Evaluation ist der fehlende vollständige Stadiontest. Dieser konnte wegen Bauarbeiten im Lakers-Stadion nicht wie geplant durchgeführt werden, da der Videowürfel vom Strom getrennt war. Dadurch konnten reale Umgebungsfaktoren nur eingeschränkt bewertet werden. Für eine produktive Weiterführung sollte deshalb ein Test unter realen Stadionbedingungen durchgeführt werden.

Für den dauerhaften produktiven Einsatz sollten insbesondere Rollenmodell, App-Integration, Scoreboard-Funktionen, Monitoring und Stadionvalidierung ergänzt werden.

11 Schlussfolgerungen und Ausblick

Dieses Kapitel fasst die wichtigsten Erkenntnisse der Arbeit zusammen. Es beurteilt die Zielerreichung, ordnet die Ergebnisse ein, reflektiert die Grenzen der Umsetzung und beschreibt mögliche nächste Schritte für eine produktive Weiterführung.

11.1 Fazit der Arbeit

Die Arbeit führte von der Analyse der Ausgangslage über die Konzeption bis zur Umsetzung eines funktionsfähigen MVP für ein interaktives Pausenspiel der SCRJ Lakers. Im Zentrum stand nicht nur eine einzelne Spielidee, sondern eine vollständige technische Lösung für Vorbereitung, Durchführung und Auswertung eines Stadion-Spiels.

Das Ergebnis ist ein MVP, der Administrationsbereich, mobile Teilnehmendenansicht, Präsentationsansicht, Backend und Deployment miteinander verbindet. Damit konnte gezeigt werden, wie Stadionpublikum, Spielleitung und Grossbildschirm in einem gemeinsamen digitalen Ablauf interagieren können.

Die Umsetzung bestätigt, dass ein solches Pausenspiel mit der gewählten Architektur technisch realisierbar ist. Gleichzeitig zeigt die Evaluation, welche Aspekte für eine produktive Weiterführung noch vertieft werden müssen. Das betrifft vor allem den realen Stadionbetrieb, die Integration in bestehende Lakers-Systeme und den Ausbau von administrativen und betrieblichen Funktionen.

Damit das System in der kommenden Saison pilotiert werden kann, muss zudem die AWS-Umgebung eingerichtet werden. Konkret betrifft dies das Deployment auf dem Account der SCRJ Lakers, die Einbindung der vereinseigenen Domain sowie die Konfiguration des Mailversands über eine offizielle E-Mail-Adresse des Vereins.

11.2 Beantwortung der Forschungsfragen

Die in der Einleitung formulierten Forschungsfragen lassen sich auf Basis von Analyse, Umsetzung und Evaluation wie folgt beantworten.

11.2.1 Forschungsfrage 1: Geeignetes Spielkonzept und Interaktion

Für ein interaktives Pausenspiel im Stadion eignet sich ein quizbasiertes Konzept am besten. Die Bewertung der untersuchten Spielideen zeigte, dass das Stadion-Quiz die Anforderungen der Lakers am ausgewogensten erfüllt. Ausschlaggebend waren vor allem die kurze Erklärbarkeit, die breite Zielgruppe, die technische Umsetzbarkeit mit der vorhandenen Infrastruktur und die Möglichkeit, einen nachvollziehbare gewinnende Person zu bestimmen.

Die Interaktion muss so gestaltet sein, dass jede Zielgruppe nur die für sie relevanten Aufgaben sieht. Zuschauende nehmen über die mobile Ansicht teil, die Spielleitung steuert den Ablauf im Administrationsbereich und der Videowürfel zeigt den gemeinsamen Spielstand für das Stadion. Diese Aufteilung reduziert die Komplexität während der kurzen Drittelpause und ermöglicht trotzdem einen synchronisierten Ablauf zwischen Publikum, Spielleitung und Präsentationsscreen.

11.2.2 Forschungsfrage 2: Erfüllung der Anforderungen durch den MVP

Der MVP zeigt, dass die gewählten Technologien für die Kernanforderungen geeignet sind. Die Anwendung deckt den vollständigen Ablauf von Vorbereitung, Teilnahme, Präsentation, Steuerung und Auswertung ab. Backend, Datenbank, mobile Teilnehmendenansicht, Präsentationsansicht und Administrationsbereich bilden zusammen eine lauffähige Grundlage für einen Pilotbetrieb.

Die Evaluation bestätigt, dass die wichtigsten funktionalen und nichtfunktionalen Anforderungen für den MVP erreicht wurden. Besonders relevant sind der geschützte Administrationsbereich, der einfache Spielbeitritt, die zuverlässige Antwortverarbeitung, die Persistenz der Resultate und das ausreichende Lastverhalten im erwarteten Nutzungsbereich. Damit beantwortet der MVP die technische Machbarkeit positiv, jedoch nur innerhalb des bewusst gesetzten Projektumfangs.

Für eine produktive Weiterführung bleiben vor allem die Validierung im echten Stadionbetrieb, die Integration in die Lakers-App und der Ausbau betrieblicher Funktionen offen. Der MVP ist deshalb nicht als abgeschlossenes Produkt zu verstehen, sondern als belastbare Grundlage, auf der ein Pilotbetrieb und die anschliessende Produktivierung aufbauen können.

11.3 Kritische Würdigung

Die Arbeit erreichte das Ziel eines funktionsfähigen MVP, dennoch bestehen klare Grenzen. Die wichtigste Einschränkung betrifft den fehlenden vollständigen Test im Stadion. Der Test war ursprünglich mit dem Videowürfel geplant, konnte wegen Bauarbeiten im Lakers-Stadion jedoch nicht durchgeführt werden, da der Videowürfel vom Strom getrennt war.

Der Ersatztest auf einem Computerdisplay mit gleicher Auflösung erlaubte eine technische Prüfung der Darstellung. Er konnte reale Stadionfaktoren wie Distanz, Lichtverhältnisse, Ablenkung, Netzqualität und Geräuschpegel jedoch nicht vollständig abbilden. Rückblickend hätte der Test der Mockups zudem stärker vergleichend angelegt werden können. Dabei hätten unterschiedliche Schriftgrössen und Designvarianten parallel getestet werden können, um die Lesbarkeit und Wirkung der Darstellung besser gegeneinander abzuwägen. Diese Faktoren sind für eine Live-Anwendung relevant und müssen vor einem produktiven Einsatz nochmals geprüft werden.

Eine weitere Einschränkung ist die fehlende direkte Integration in die Lakers-App. Der MVP verwendet einen mobilen Link beziehungsweise QR-Code und identifiziert Teilnehmende über Sitzungstoken. Damit ist der Spielbeitritt möglich, die ursprünglich angedachte Einbettung in den bestehenden App-Kontext der Lakers bleibt aber offen.

11.4 Ausblick

Der wichtigste nächste Schritt ist ein Test unter echten Stadionbedingungen. Dabei sollten Videowürfel, Smartphone-Nutzung, Netzqualität, Sichtbarkeit, Lichtverhältnisse und der Ablauf während einer realen Pause gemeinsam geprüft werden. Erst dadurch lässt sich beurteilen, wie sich der MVP im tatsächlichen Stadionkontext unter realen Umgebungsbedingungen verhält.

Für eine produktive Weiterführung sollte die Integration in die Lakers-App durch den Hersteller geprüft werden. Dadurch könnten Reichweite, Identifikation der Teilnehmenden und Einstieg in das Spiel verbessert werden. Alternativ oder ergänzend wäre ein Beitritt über einen ausreichend gross dargestellten Code zu prüfen, falls QR-Codes im Stadion nicht für alle Zuschauenden zuverlässig nutzbar sind.

Technisch bietet die bestehende Architektur eine Grundlage für weitere Ausbauschritte. Dazu gehören ein detailliertes Rollenmodell, Monitoring, saisonübergreifende Statistiken, ein öffentliches Scoreboard und weitere Spielmodi. Auch die Sponsorenintegration kann ausgebaut werden, beispielsweise durch flexiblere Platzierung von Sponsoring-Elementen oder spielmodusspezifische Darstellungen.

Bei höheren erwarteten Teilnahmezahlen sollte die Infrastruktur erneut unter Last geprüft werden. Die gewählte AWS-Architektur ermöglicht Skalierung, die passende Dimensionierung muss jedoch anhand realer Nutzungszahlen und weiterer Lasttests bestimmt werden.

12 Glossary und Abkürzungen

Begriff	Bedeutung
ACU	Aurora Compute Unit, von AWS definierte Recheneinheit zur Abrechnung von Datenbankleistung
Administrationsbereich	Steuerungsoberfläche, in der Spielinhalte vorbereitet, Spiele gestartet und Ergebnisse verwaltet werden
Anwendung	Die umgesetzte Softwarelösung mit Frontend, Backend, Datenhaltung und Deployment
API	Application Programming Interface, Schnittstelle, über die Softwaresysteme miteinander kommunizieren
AWS	Amazon Web Services, Cloud-Plattform für Infrastruktur- und Plattformdienste
CI/CD	Continuous Integration / Continuous Deployment, automatisierte Prozesse zum Prüfen, Bauen und Bereitstellen von Software
DTO	Data Transfer Object, Objekt zur strukturierten Übertragung und Validierung von Daten zwischen Systemteilen
EC2	Elastic Compute Cloud, Cloud-Computing-Dienst von AWS für virtuelle Server
ECR	Elastic Container Registry, AWS-Dienst zur Speicherung und Verwaltung von Containerimages
ECS	Elastic Container Service, AWS-Dienst zur Ausführung und Verwaltung containerisierter Anwendungen
End-to-End-Test	Test eines vollständigen Ablaufs über mehrere Systemteile hinweg
Healthcheck	Technische Prüfung, ob ein Dienst lauffähig ist und seine Abhängigkeiten erreichbar sind
IAM	Identity and Access Management, AWS-Dienst zur Verwaltung von Benutzenden, Rollen und Zugriffsrechten
IIHF	International Ice Hockey Federation, internationaler Eishockeyverband und Ausrichter der Eishockey-Weltmeisterschaft
Joi	Bibliothek zur Validierung von Konfigurationswerten und Datenstrukturen
JSONB	Binary JSON, PostgreSQL-Datentyp zur effizienten Speicherung und Abfrage strukturierter JSON-Daten
JWT	JSON Web Token, tokenbasiertes Format zur Übertragung von Authentifizierungs- und Autorisierungsinformationen
Linting	Statische Codeprüfung zur Erkennung von Regelverstössen, Formatierungsproblemen und potenziellen Fehlern
Loadbalancer	Dienst zur Verteilung eingehender HTTP- und HTTPS-Anfragen auf Zielsysteme

Begriff	Bedeutung
Deployment	Prozess, mit dem die Anwendung gebaut, konfiguriert und in einer ausführbaren Umgebung betrieben wird
Lasttest	Prüfung des Systemverhaltens unter vielen gleichzeitigen Zugriffen
mobile Teilnehmendenansicht	Smartphone-Oberfläche, über die Teilnehmende einem Spiel beitreten, Antworten abgeben und Ergebnisrückmeldungen erhalten
Mockup	Visueller Entwurf einer Bedienoberfläche zur frühen Prüfung von Aufbau, Inhalt und Bedienung
MVP	Minimum Viable Product, minimal funktionsfähige Version des Produkts, die in dieser Arbeit als finales Ergebnis umgesetzt und evaluiert wird
NFR	Non-Functional Requirement, Abkürzung für eine nichtfunktionale Anforderung an ein System, zum Beispiel Performance, Sicherheit oder Wartbarkeit
Nginx	Webserver, der in der Anwendung den Frontend-Build ausliefert
PRTG	Paessler Router Traffic Grapher, Monitoring-Software zur Überwachung von Netzwerken und Systemen
Präsentationsansicht	Softwareansicht für die Darstellung des Spielzustands auf dem Videowürfel
Rate Limiting	Begrenzung der Anzahl Anfragen pro Client oder Zeitraum zum Schutz vor Überlastung und Missbrauch
RDS	Relational Database Service, Datenbankdienst von AWS für den Betrieb relationaler Datenbanken
S3	Simple Storage Service, Cloud-Speicherdienst von AWS für Dateien und Objekte
Secret	Sensibler Konfigurationswert wie Passwort, API-Schlüssel oder JWT-Schlüssel
SES	Simple Email Service, AWS-Dienst zum Versand von E-Mails
Socket.IO	Bibliothek für Echtzeitkommunikation mit Funktionen wie Räumen und automatischer Wiederverbindung
SSM	Systems Manager, AWS-Dienst zur Verwaltung von Konfigurationen, Parametern und Betriebsaufgaben
Testabdeckung	Anteil des Codes, der während automatisierter Tests ausgeführt wird
Testdouble	Ersatzobjekt in automatisierten Tests, das eine reale Abhängigkeit kontrolliert nachbildet
TypeORM	Object-Relational Mapper für TypeScript und JavaScript
Videowürfel	Stadionanzeige der SCRJ Lakers für grossflächige Spielinformationen im Publikumskontext

Begriff	Bedeutung
VPC	Virtual Private Cloud, virtuelles Netzwerk innerhalb einer Cloud-Umgebung
WebSocket	Kommunikationsprotokoll für bidirektionale Echtzeitkommunikation zwischen Client und Server

13 Abbildungsverzeichnis

Abbildung 1 Zentrale Ansichten des MVP: Administrationsbereich
Quelle: Eigene Aufnahme 3
Abbildung 2 Zentrale Ansichten des MVP: Präsentationsansicht und mobile Teilnehmendenansicht
Quelle: Eigene Aufnahme 4
Abbildung 3 Zeitlicher Projektablauf nach Kalenderwochen 9
Abbildung 4 Mindmap
Quelle: Eigene Darstellung, erstellt mit Xmind 12
Abbildung 5 Sketch Smartphone Sicht vom offensiven & defensiven Team
Quelle: Eigene Darstellung, erstellt mit Figma 13
Abbildung 6 Sketch Videowürfel Sicht
Quelle: Eigene Darstellung, erstellt mit Figma 13
Abbildung 7 Screenshot originales Flappy Bird
Quelle: (DotGears, o. J.) 15
Abbildung 8 Sketch Sicht Videowürfel
Quelle: Eigene Darstellung, erstellt mit Figma 16
Abbildung 9 Sketch Sicht Smartphone
Quelle: Eigene Darstellung, erstellt mit Figma 17
Abbildung 10 Sketch Videowürfel Sicht
Quelle: Eigene Darstellung, erstellt mit Figma 19
Abbildung 11 Sketch Kreuzworträtsel Videowürfel Sicht
Quelle: Eigene Darstellung, erstellt mit Figma 20
Abbildung 12 Sketch Bilderrätsel Videowürfel Sicht
Quelle: Eigene Darstellung, erstellt mit Figma 21
Abbildung 13 Sketch Smartphone Sicht
Quelle: Eigene Darstellung, erstellt mit Figma 23
Abbildung 14 Anwendungsfall Diagram
Quelle: Eigene Darstellung 39
Abbildung 15 Entwurfsansicht des Anmeldebildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma 54
Abbildung 16 Entwurfsansicht des Anmeldebildschirms für alle Spielmodi in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma 55
Abbildung 17 Entwurfsansicht der Hauptansicht des Administrationsbereichs
Quelle: Eigene Darstellung, erstellt mit Figma 55
Abbildung 18 Foto vom Test der Entwurfsansichten im Stadion
Quelle: Eigene Aufnahme 57
Abbildung 19 Überarbeitete Entwurfsansicht des Anmeldebildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma 57
Abbildung 20 Überarbeitete Entwurfsansicht des Fragebildschirms für den Spielmodus Kreuzworträtsel
Quelle: Eigene Darstellung, erstellt mit Figma 58

Abbildung 21 Überarbeitete Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma 58
Abbildung 22 Komponentendiagramm des Frontends 64
Abbildung 23 Klassendiagramm des Backends 66
Abbildung 24 Datenbankstruktur
Quelle: Eigene Abbildung, erstellt mit draw.io 69
Abbildung 25 Übersicht des Administrationsbereichs mit Spielvorlagen und Spielverlauf
Quelle: Eigene Aufnahme 79
Abbildung 26 Bearbeitung einer Quizfrage im Administrationsbereich
Quelle: Eigene Aufnahme 80
Abbildung 27 Präsentationsansicht im Wartezustand eines gestarteten Durchlaufs
Quelle: Eigene Aufnahme 81
Abbildung 28 mobile Teilnehmendenansicht im Wartezustand
Quelle: Eigene Aufnahme 82
Abbildung 29 Ablauf der GitLab-CI/CD-Pipeline für Erstellung, Deployment und Kostensteuerung
Quelle: Eigene Abbildung 89
Abbildung 30 AWS-Deployment-Architektur
Quelle: Eigene Abbildung, erstellt mit draw.io 91
Abbildung 31 Visualisierung der API-Antwortzeiten mit minimalen AWS-Ressourcen
Quelle: Eigene Abbildung, erstellt mit Canva 103
Abbildung 32 Visualisierung der API-Antwortzeiten mit verdoppelten AWS-Ressourcen
Quelle: Eigene Abbildung, erstellt mit Canva 103
Abbildung 33 Ablaufdiagramm zur Verwaltung von Admininhalten
Quelle: Eigene Abbildung, erstellt mit draw.io 131
Abbildung 34 Sequenzdiagramm einer Live-Spielrunde
Quelle: Eigene Abbildung, erstellt mit Mermaid 132
Abbildung 35 Entwurfsansicht des Anmeldebildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma 133
Abbildung 36 Entwurfsansicht des Anmeldebildschirms für alle Spielmodi in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma 133
Abbildung 37 Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma 134
Abbildung 38 Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma 134
Abbildung 39 Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi in der mobilen Teilnehmendenansicht, wenn die teilnehmende

Person nicht gewonnen hat
Quelle: Eigene Darstellung, erstellt mit Figma 135
Abbildung 40 Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Quiz
Quelle: Eigene Darstellung, erstellt mit Figma 135
Abbildung 41 Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Quiz in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma 136
Abbildung 42 Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Quiz angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma 136
Abbildung 43 Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Quiz in der mobilen Teilnehmendenansicht angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma 137
Abbildung 44 Entwurfsansicht des Anmeldebildschirms für den Spielmodus Kreuzworträtsel
Quelle: Eigene Darstellung, erstellt mit Figma 137
Abbildung 45 Entwurfsansicht des Anmeldebildschirms für den Spielmodus Kreuzworträtsel in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma 138
Abbildung 46 Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Kreuzworträtsel angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma 138
Abbildung 47 Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Kreuzworträtsel in der mobilen Teilnehmendenansicht angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma 139
Abbildung 48 Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Bilderrätsel
Quelle: Eigene Darstellung, erstellt mit Figma 139
Abbildung 49 Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Bilderrätsel in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma 140
Abbildung 50 Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Bilderrätsel angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma 140
Abbildung 51 Entwurfsansicht des Bildschirms während die Antwort des Spielmodus Bilderrätsel in der mobilen Teilnehmendenansicht angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma 141
Abbildung 52 Entwurfsansicht der Hauptansicht des Administrationsbereichs
Quelle: Eigene Darstellung, erstellt mit Figma 141

Abbildung 53 Entwurfsansicht zum Erstellen eines Spiels mit dem Spielmodus Quiz im Administrationsbereich
Quelle: Eigene Darstellung, erstellt mit Figma 142
Abbildung 54 Entwurfsansicht zum Erstellen eines Spiels mit dem Spielmodus Kreuzworträtsel im Administrationsbereich
Quelle: Eigene Darstellung, erstellt mit Figma 143
Abbildung 55 Entwurfsansicht zum Erstellen eines Spiels mit dem Spielmodus Bilderrätsel im Administrationsbereich
Quelle: Eigene Darstellung, erstellt mit Figma 144
Abbildung 56 Foto vom Test der Entwurfsansichten im Stadion
Quelle: Eigene Aufnahme 145
Abbildung 57 Foto vom Test der Entwurfsansichten im Stadion
Quelle: Eigene Aufnahme 145
Abbildung 58 Überarbeitete Entwurfsansicht des Anmeldebildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma 146
Abbildung 59 Überarbeitete Entwurfsansicht des Anmeldebildschirms für den Spielmodus Kreuzworträtsel
Quelle: Eigene Darstellung, erstellt mit Figma 146
Abbildung 60 Überarbeitete Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Kreuzworträtsel angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma 147
Abbildung 61 Überarbeitete Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Bilderrätsel
Quelle: Eigene Darstellung, erstellt mit Figma 147
Abbildung 62 Überarbeitete Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Bilderrätsel angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma 148
Abbildung 63 Überarbeitete Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma 148
Abbildung 64 Administrationsbereich: Anmeldeansicht für Administrierende.
Quelle: Eigene Aufnahme 149
Abbildung 65 Administrationsbereich: Übersicht mit Spielvorlagen, Filtern und Spielverlauf.
Quelle: Eigene Aufnahme 149
Abbildung 66 Administrationsbereich: Formular zum Erstellen einer neuen Spielvorlage.
Quelle: Eigene Aufnahme 150
Abbildung 67 Administrationsbereich: Bearbeitung einer Quiz-Spielvorlage.
Quelle: Eigene Aufnahme 150
Abbildung 68 Administrationsbereich: Bearbeitung einer Bilderquiz-Spielvorlage.
Quelle: Eigene Aufnahme 151

Abbildung 69 Administrationsbereich: Bearbeitung einer Kreuzworträtsel-Spielvorlage.
Quelle: Eigene Aufnahme 151
Abbildung 70 Administrationsbereich: Verwaltung der verfügbaren Spielmodi.
Quelle: Eigene Aufnahme 152
Abbildung 71 Administrationsbereich: Verwaltung der Sponsoren und Sponsorendaten.
Quelle: Eigene Aufnahme 152
Abbildung 72 Administrationsbereich: Verwaltung der Administrationskonten.
Quelle: Eigene Aufnahme 153
Abbildung 73 Administrationsbereich: Statistikansicht eines abgeschlossenen Quiz-Durchlaufs.
Quelle: Eigene Aufnahme 153
Abbildung 74 Präsentationsansicht Quiz: Wartezustand mit QR-Code und Beitrittscode.
Quelle: Eigene Aufnahme 154
Abbildung 75 Präsentationsansicht Quiz: Anzeige einer Frage mit Antwortoptionen.
Quelle: Eigene Aufnahme 154
Abbildung 76 Präsentationsansicht Quiz: Auflösung der richtigen Antwort.
Quelle: Eigene Aufnahme 155
Abbildung 77 Präsentationsansicht Quiz: Abschlussansicht nach Spielende.
Quelle: Eigene Aufnahme 155
Abbildung 78 Präsentationsansicht Bilderquiz: Wartezustand mit QR-Code und Beitrittscode.
Quelle: Eigene Aufnahme 156
Abbildung 79 Präsentationsansicht Bilderquiz: Anzeige einer Bildfrage.
Quelle: Eigene Aufnahme 156
Abbildung 80 Präsentationsansicht Bilderquiz: Auflösung der Bildfrage.
Quelle: Eigene Aufnahme 157
Abbildung 81 Präsentationsansicht Bilderquiz: Abschlussansicht nach Spielende.
Quelle: Eigene Aufnahme 157
Abbildung 82 Präsentationsansicht Kreuzworträtsel: Wartezustand mit QR-Code und Beitrittscode.
Quelle: Eigene Aufnahme 158
Abbildung 83 Präsentationsansicht Kreuzworträtsel: Anzeige der Hinweise im Fragemodus.
Quelle: Eigene Aufnahme 158
Abbildung 84 Präsentationsansicht Kreuzworträtsel: Auflösung der Antworten.
Quelle: Eigene Aufnahme 159
Abbildung 85 Präsentationsansicht Kreuzworträtsel: Abschlussansicht nach Spielende.
Quelle: Eigene Aufnahme 159

Abbildung 86 Präsentationsansicht mit Administrations-Overlay: Quiz im Wartezustand mit Beitrittscode.
Quelle: Eigene Aufnahme 160
Abbildung 87 Präsentationsansicht mit Administrations-Overlay: Quiz im Fragemodus.
Quelle: Eigene Aufnahme 160
Abbildung 88 Präsentationsansicht mit Administrations-Overlay: Quiz in der Auflösung.
Quelle: Eigene Aufnahme 161
Abbildung 89 Präsentationsansicht mit Administrations-Overlay: Quiz nach Spielende.
Quelle: Eigene Aufnahme 161
Abbildung 90 mobile Teilnehmendenansicht Quiz: Wartezustand nach dem Beitritt.
Quelle: Eigene Aufnahme 162
Abbildung 91 mobile Teilnehmendenansicht Quiz: Auswahl einer Antwortoption.
Quelle: Eigene Aufnahme 163
Abbildung 92 mobile Teilnehmendenansicht Quiz: Rückmeldung zur eigenen Antwort.
Quelle: Eigene Aufnahme 164
Abbildung 93 mobile Teilnehmendenansicht Quiz: Ergebnisansicht nach Spielende mit eindeutiger Gewinn- oder Nicht-Gewinn-Rückmeldung.
Quelle: Eigene Aufnahme 165
Abbildung 94 mobile Teilnehmendenansicht Bilderquiz: Wartezustand nach dem Beitritt.
Quelle: Eigene Aufnahme 166
Abbildung 95 mobile Teilnehmendenansicht Bilderquiz: Auswahl einer Antwortoption zur Bildfrage.
Quelle: Eigene Aufnahme 167
Abbildung 96 mobile Teilnehmendenansicht Bilderquiz: Rückmeldung zur eigenen Antwort.
Quelle: Eigene Aufnahme 168
Abbildung 97 mobile Teilnehmendenansicht Bilderquiz: Ergebnisansicht nach Spielende mit eindeutiger Gewinn- oder Nicht-Gewinn-Rückmeldung.
Quelle: Eigene Aufnahme 169
Abbildung 98 mobile Teilnehmendenansicht Kreuzworträtsel: Wartezustand nach dem Beitritt.
Quelle: Eigene Aufnahme 170
Abbildung 99 mobile Teilnehmendenansicht Kreuzworträtsel: Eingabeansicht für die Wörter.
Quelle: Eigene Aufnahme 171

Abbildung 100 mobile Teilnehmendenansicht Kreuzworträtsel: Statische Auflösung der eingegebenen Antworten.
Quelle: Eigene Aufnahme 172
Abbildung 101 mobile Teilnehmendenansicht Kreuzworträtsel: Ergebnisansicht nach Spielende mit eindeutiger Gewinn- oder Nicht-Gewinn-Rückmeldung.
Quelle: Eigene Aufnahme 173
Abbildung 102 Aufnahme Besuchende 27.02.2026 im Lakersstadion Sektor B mit 290 gezählten Personen nach einer Minute Pause
Quelle: Eigene Aufnahme 174
Abbildung 103 Aufnahme Besuchende 27.02.2026 im Lakersstadion Sektor C mit 500 gezählten Personen nach einer Minute Pause
Quelle: Eigene Aufnahme 175
Abbildung 104 Aufnahme Besuchende 27.02.2026 im Lakersstadion Sektor B mit 231 gezählten Personen nach drei Minuten Pause
Quelle: Eigene Aufnahme 175
Abbildung 105 Aufnahme Besuchende 27.02.2026 im Lakersstadion Sektor C mit 354 gezählten Personen nach drei Minuten Pause
Quelle: Eigene Aufnahme 176
Abbildung 106 Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor B mit 307 gezählten Personen nach einer Minute Pause
Quelle: Eigene Aufnahme 177
Abbildung 107 Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor C mit 630 gezählten Personen nach einer Minute Pause
Quelle: Eigene Aufnahme 177
Abbildung 108 Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor B mit 190 gezählten Personen nach drei Minuten Pause
Quelle: Eigene Aufnahme 178
Abbildung 109 Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor C mit 460 gezählten Personen nach drei Minuten Pause
Quelle: Eigene Aufnahme 178
Abbildung 110 Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor B mit 174 gezählten Personen nach vier Minuten Pause
Quelle: Eigene Aufnahme 179
Abbildung 111 Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor C mit 446 gezählten Personen nach vier Minuten Pause
Quelle: Eigene Aufnahme 179

14 Tabellenverzeichnis

Tabelle 1 Bewertungskriterium: Sponsorenintegration 25
Tabelle 2 Bewertung Sponsorenintegration 25
Tabelle 3 Bewertungskriterium: Einbindung in die Lakers-App 26
Tabelle 4 Bewertung Einbindung in die Lakers-App 26
Tabelle 5 Bewertungskriterium: Marketingaufwand 27
Tabelle 6 Bewertung Marketingaufwand 27
Tabelle 7 Bewertungskriterium: Anzahl der Teilnehmenden 28
Tabelle 8 Bewertung Anzahl der Teilnehmenden 28
Tabelle 9 Bewertungskriterium: Zielgruppeninklusion 28
Tabelle 10 Bewertung Zielgruppeninklusion 29
Tabelle 11 Bewertungskriterium: Verständlichkeit des Spielkonzepts 29
Tabelle 12 Bewertung Verständlichkeit des Spielkonzepts 29
Tabelle 13 Bewertungskriterium: Scoreboard-Integration 30
Tabelle 14 Bewertung Scoreboard-Integration 31
Tabelle 15 Bewertungskriterium: Technische Machbarkeit 32
Tabelle 16 Bewertung Technische Machbarkeit 32
Tabelle 17 Bewertungskriterium: Entwicklungsaufwand 33
Tabelle 18 Bewertung Entwicklungsaufwand 33
Tabelle 19 Bewertungskriterium: Beinhaltet keine No-Gos 34
Tabelle 20 Bewertung: Beinhaltet keine No-Gos 34
Tabelle 21 Bewertungskriterium: Spieldauer 34
Tabelle 22 Bewertung Spieldauer 35
Tabelle 23 Gesamtbewertung der Spielideen 35
Tabelle 24 Auswertung der Publikumszahlen in den Sektoren B und C während der zweiten Pause 36
Tabelle 25 Bewertungsskala für den Mitmachwert des Pausenspiels 37
Tabelle 26 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC01 abgeleitet wurden 40
Tabelle 27 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC02 abgeleitet wurden 40
Tabelle 28 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC03 abgeleitet wurden 40
Tabelle 29 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC04 abgeleitet wurden 41
Tabelle 30 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC05 abgeleitet wurden 41
Tabelle 31 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC11 abgeleitet wurden 42
Tabelle 32 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC12 abgeleitet wurden 43
Tabelle 33 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC13 abgeleitet wurden 43

Tabelle 34 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC21 abgeleitet wurden 44
Tabelle 35 Liste aller funktionalen Anforderungen, die aus dem Anwendungsfall UC22 abgeleitet wurden 44
Tabelle 36 Funktionale Aspekte der nichtfunktionalen Anforderungen 46
Tabelle 37 Benutzbarkeitsaspekte der nichtfunktionalen Anforderungen 47
Tabelle 38 Zuverlässigkeitsaspekte der nichtfunktionalen Anforderungen 48
Tabelle 39 Leistungsaspekte der nichtfunktionalen Anforderungen 49
Tabelle 40 Wartbarkeitsaspekte der nichtfunktionalen Anforderungen 50
Tabelle 41 Vergleich der evaluierten Betriebsoptionen 86
Tabelle 42 Validierung der funktionalen Anforderungen 96
Tabelle 43 Validierung der funktionalen Anforderungen 97
Tabelle 44 Validierung der nichtfunktionalen Anforderungen 99
Tabelle 45 Validierung der nichtfunktionalen Anforderungen 100
Tabelle 46 Validierung der nichtfunktionalen Anforderungen 101
Tabelle 47 Zusammenfassung des API-Lasttests mit minimalen AWS-Ressourcen 102
Tabelle 48 Zusammenfassung des API-Lasttests mit verdoppelten AWS-Ressourcen 103
Tabelle 49 Zusammenfassung der Usability-Umfrage mit vier Testpersonen . . 105
Tabelle 50 Branch Prefixes 187
Tabelle 51 Commit Types 188

15 Literaturverzeichnis

Ante, L., Saggu, A., Schellinger, B., & Wazinski, F.-P. (2024). Voting participation and engagement in blockchain-based fan tokens. Electronic Markets, 34, Article 26. https://doi.org/10.1007/s12525-024-00709-z

Barre, A. (o. J.). eslint-plugin-react-refresh. GitHub. Abgerufen 28. Mai 2026, von https://github.com/ArnaudBarre/eslint-plugin-react-refresh#vite-config

Bundesgesetz über den Datenschutz (Datenschutzgesetz, DSG), SR 235.1 (2022). https://www.fedlex.admin.ch/eli/cc/2022/491/de

Burchett, Michael H. (2022,). The Wave (mass guesture). EBSCO Research Starters; EBSCO Information Services. https://www.ebsco.com/research-starters/sports-and-leisure/wave-mass-guesture

DotGears. (o. J.). Flappy Bird. Abgerufen 4. März 2026, von https://dotgears.com/games/flappybird

Eska, J., & Bichsel, M. (2021, April 7). Kanban - richtig eingesetzt. Wirtschaftsinformatik Blog; Fachhochschule Nordwestschweiz FHNW. https://www.fhnw.ch/plattformen/iwi/2021/04/07/kanban-richtig-eingesetzt/

Giorgio, P., Reichheld, A., Deweese, C., & Ebb, S. (o. J.). The stadium experience: Keeping sports fans engaged and loyal. Deloitte Consulting LLP. Abgerufen 16. März 2026, von https://www.deloitte.com/content/dam/assets-zone3/us/en/docs/industries/technology-media-telecommunications/2024/us-tmt-stadium-experience-keeping-sports-fans-engaged-loyal.pdf

International Organization for Standardization. (2018, März). ISO 9241-11:2018 Ergonomics of human-system interaction - Part 11: Usability: Definitions and concepts. ISO. https://www.iso.org/standard/63500.html

Martins, F., França, C., Paixão, P., Martinho, D. V., Campos, P., Gouveia, B., Lopes, H., Ihle, A., Marques, E., & Gouveia, É. R. (2023). Emerging technologies to promote fans interaction in football events: A systematic review. Advances in Human-Computer Interaction, 2023, 1–13. https://doi.org/10.1155/2023/6667260

Meta Platforms, Inc. (o. J.). eslint-plugin-react-hooks. React. Abgerufen 28. Mai 2026, von https://react.dev/reference/eslint-plugin-react-hooks#recommended-rules

Musa, K., & Alkhateeb, J. (2013). Quality Model Based on COTS Quality Attributes. International Journal of Software Engineering & Applications, 4(1), 1–8. https://doi.org/10.5121/ijsea.2013.4101

OpenJS Foundation and ESLint contributors. (o. J.). Configuration migration guide. ESLint. Abgerufen 28. Mai 2026, von https://eslint.org/docs/latest/use/configure/migration-guide#predefined-and-shareable-configs

Twitch Plays Pokémon. (o. J.). Twitch Plays Pokémon. Abgerufen 4. März 2026, von https://twitchplayspokemon.tv/

Vite contributors. (o. J.). Vite React TypeScript template ESLint configuration. GitHub. Abgerufen 28. Mai 2026, von https://github.com/vitejs/vite/blob/main/packages/create-vite/template-react-ts/eslint.config.js

16 Hilfsmittelverzeichnis

Im Rahmen dieser Arbeit wurden folgende KI-Assistenten und Tools eingesetzt:

Aufgabenbereich	**Tools**
Recherche	Google, ChatGPT, Claude, YouTube
Datenanalyse und Visualisierung	Drawio, Mermaid, Canva
Prototyping	Figma, Goodnotes
Coding	Visual Studio Code, JetBrains, ChatGPT, Claude, React, NestJS, PostgreSQL
Texterstellung, Textoptimierung, Rechtschreibe- und Grammatikprüfung	typst, Copilot, ChatGPT, Claude, Visual Studio Code
Zusammenarbeit und Projektmanagement	Outlook, Teams, GitLab, YouTrack, WhatsApp, typst
DevOps	Docker, GitLab, AWS

Part III

Anhang A: Technische Artefakte

Inhaltsverzeichnis Anhang A

A.1 Initial Project Proposal 129
A.2 Diagramme 131
A.3 Mockups 133
A.3.1 Allgemeine Mockups 133
A.3.2 Entwurfsansichten des Spielmodus Quiz 135
A.3.3 Entwurfsansichten des Spielmodus Kreuzworträtsel 137
A.3.4 Entwurfsansichten des Spielmodus Bilderrätsel 139
A.3.5 Entwurfsansichten des Administrationsbereichs 141
A.3.6 Test im Stadion 145
A.3.7 Entwurfsansichten Version 2 146
A.4 Screenshots der Anwendung 149
A.4.1 Administrationsbereich 149
A.4.2 Präsentationsansicht Quiz 154
A.4.3 Präsentationsansicht Bilderquiz 156
A.4.4 Präsentationsansicht Kreuzworträtsel 158
A.4.5 Präsentationsansicht mit Administrations-Overlay 160
A.4.6 mobile Teilnehmendenansicht Quiz 162
A.4.7 mobile Teilnehmendenansicht Bilderquiz 166
A.4.8 mobile Teilnehmendenansicht Kreuzworträtsel 170
A.5 Publikumszahlenauswertung 174
A.5.1 Spiel 1: Lakersstadion, 27.02.2026 174
A.5.2 Spiel 2: Lakersstadion, 05.03.2026 177
A.6 Umfrageergebnisse Usertest 180
A.7 Qualitätsstandart 186

A.1 Initial Project Proposal

Interaktive Spiele für die Pausenunterhaltung bei den Rapperswil-Jona Lakers

Beteiligte Personen

Simeon Rhyner, simeon.rhyner@ost.ch

Jeriel Neal Frei, jeriel.frei@ost.ch

Betreuer dieser Arbeit
Prof. Dr.-Ing. Frieder Loch; frieder.loch@ost.ch

Experte
Dr. Simon Philipp

Problembeschreibung

Im Frühlingssemester 2026 werden neue digitale Möglichkeiten zur Fan-Interaktion für die Lakers untersucht. Im Zentrum steht die Entwicklung innovativer Spielformate, die während den Pausen bei Spielen genutzt werden können. Diese Spiele sollen sowohl auf dem Videowürfel als auch auf den Mobilgeräten der Fans spielbar sein und gleichzeitig die Nutzung der neuen Lakers-App fördern.

Die Bachelorarbeit baut auf dieser Ausgangslage auf und verfolgt das Ziel, geeignete Spielkonzepte zu erarbeiten, diese prototypisch umzusetzen und in der Eishalle zu evaluieren. Dabei stehen motivierende, leicht verständliche und gemeinschaftlich spielbare Interaktionsformate im Fokus, die das Fan-Erlebnis stärken und den Stadionbesuch attraktiver gestalten.

Formulierung eines konkreten Auftrags

Die Arbeit soll die folgenden Aspekte behandeln. Nach Absprache kann im Verlauf der Arbeit vom beschriebenen Auftrag abgewichen werden:

- **Analyse der Anforderungen.** Erhebung und Definition der funktionalen, technischen und organisatorischen Anforderungen. Basis bilden Gespräche mit den Lakers, Beobachtungen in der Arena sowie eine Analyse der Rahmenbedingungen.
- **Recherche und Abgrenzung.** Untersuchung bestehender Fan-Engagement-Spiele, stadiontauglicher Interaktionsformate und technologischer Lösungen. Identifikation von Trends, Lücken und Potenzialen für neue Formate.
- **Nutzerzentrierte Analyse.** Gestaltung der Spielmechaniken und Interaktionsabläufe nach einem nutzerzentrierten Vorgehen. Berücksichtigung unterschiedlicher Zielgruppen sowie der besonderen Nutzungssituation während einer kurzen Stadionpause.
- **Systemarchitektur und Technologieauswahl.** Definition einer geeigneten Systemarchitektur für die Spiele sowie Auswahl geeigneter Technologien. Dazu gehören u. a. Überlegungen zum Zusammenspiel zwischen App, Backend, Videowürfel, Mobilgeräten und Live-Synchronisierung.
- **Prototypische Umsetzung und Evaluation.** Entwicklung eines funktionsfähigen Prototyps. Durchführung von Tests in einer realistischen Umgebung – idealerweise in der Eishalle oder mit einer grösseren Testgruppe.

- **Kritische Reflexion.** Bewertung des Vorgehens, der Konzeptentscheidungen, der technischen Umsetzung sowie der Praxistauglichkeit der entwickelten Spiele. Diskussion möglicher Verbesserungen oder Erweiterungen für einen späteren produktiven Einsatz.

Für das Projekt ist eine geeignete Methode des Projektmanagements zu wählen (z. B. agile Methoden, hybride Ansätze oder ein klassisches Vorgehen). Arbeitspakete, Meilensteine und eine realistische Zeitplanung sind zu dokumentieren und während des Projekts laufend zu aktualisieren.

Umfang und Form der erwarteten Resultate

Die Ergebnisse der Arbeit (Quellcode der Software inkl. Dokumentation, sowie der schriftliche Projektbericht) werden den Projektbeteiligten zur weiteren Nutzung zur Verfügung gestellt.

Bei der Erstellung des Berichts werden die üblichen Qualitätsstandards für die Erstellung wissenschaftlicher Arbeiten angewendet. So müssen z.B. alle Quellen (z.B. für Definitionen) konkret und spezifisch nachgewiesen werden. Dies gilt auch für alle Abbildungen, sofern sie nicht selbst erstellt wurden. Quellennachweise erfolgen im Literaturverzeichnis im APA-Stil.

Bei der Erstellung des Berichts werden die Empfehlungen des Bundes zur geschlechtergerechten Sprache berücksichtigt.

Anfangs- und Abgabetermin

- Start der Bearbeitung: **Montag, 16. Februar 2026**
- Abgabe: **Freitag, 12. Juni 2026 (17:00 Uhr)**

Zulässige Hilfsmittel und weitere Betreuung

Alle verwendeten Hilfsmittel müssen in der Arbeit aufgeführt werden. Die Nutzung von KI-Werkzeugen ist entsprechend der Leitlinie für den KI-Einsatz in der Lehre zulässig. Die Betreuung erfolgt durch die genannte Betreuungsperson. Es werden wöchentliche Beratungstermine vereinbart und von den Studierenden protokolliert.

A.2 Diagramme

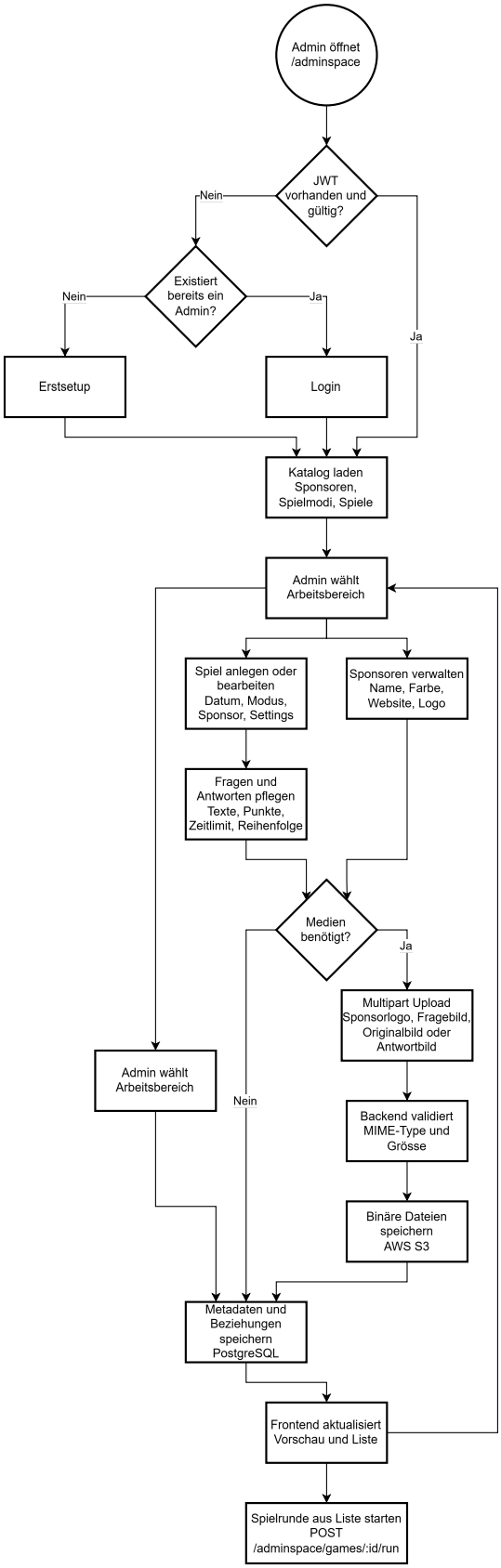


Abbildung 33: Ablaufdiagramm zur Verwaltung von Admininhalten
Quelle: Eigene Abbildung, erstellt mit draw.io

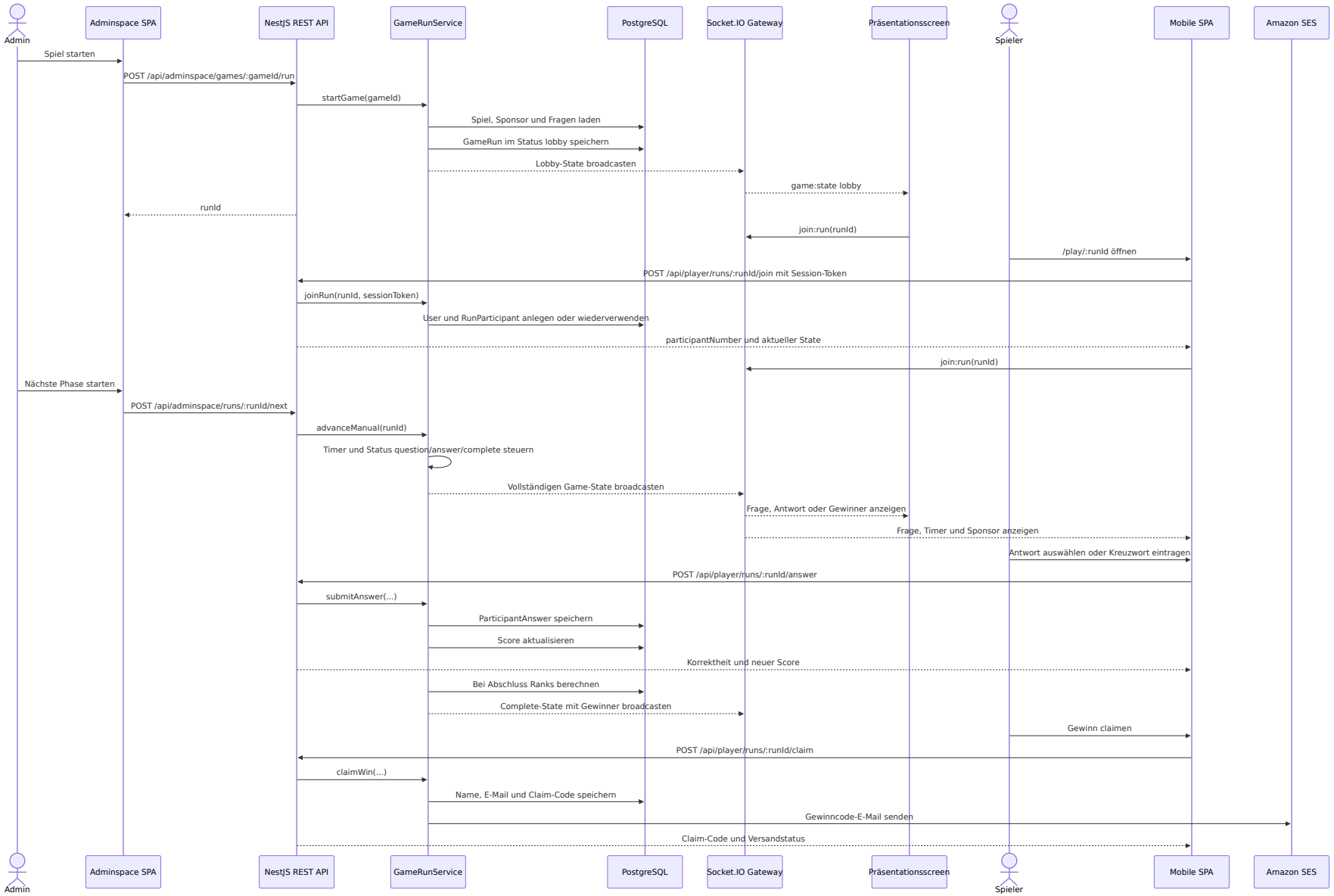


Abbildung 34: Sequenzdiagramm einer Live-Spielrunde

Quelle: Eigene Abbildung, erstellt mit Mermaid

A.3 Mockups

A.3.1 Allgemeine Mockups

In diesem Kapitel werden die vollständigen Entwurfsansichten aufgeführt. Sie ergänzen die gekürzte Darstellung im Hauptteil und dokumentieren die Varianten der Spielmodi.

Anmeldephase:



Abbildung 35: Entwurfsansicht des Anmeldebildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma

Anmeldephase mobile Teilnehmendenansicht:

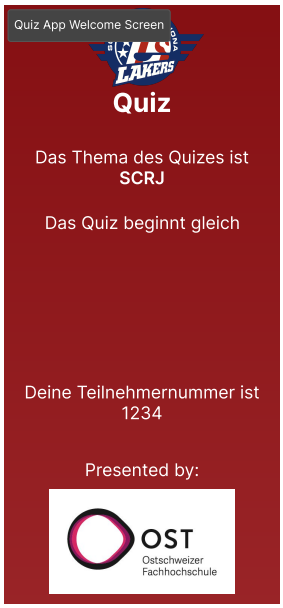


Abbildung 36: Entwurfsansicht des Anmeldebildschirms für alle Spielmodi in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma

Auswertungsphase:



Abbildung 37: Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma

Auswertungsphase Gewinn mobile Teilnehmendenansicht:

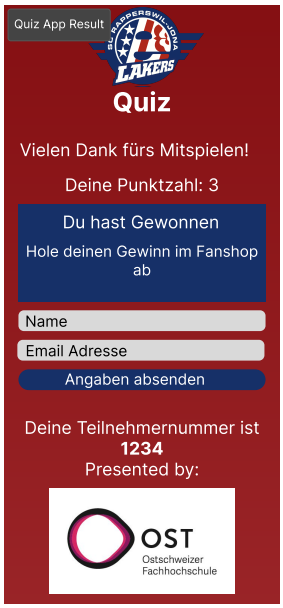


Abbildung 38: Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma

Auswertungsphase mobile Teilnehmendenansicht:



Abbildung 39: Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi in der mobilen Teilnehmendenansicht, wenn die teilnehmende Person nicht gewonnen hat
Quelle: Eigene Darstellung, erstellt mit Figma

A.3.2 Entwurfsansichten des Spielmodus Quiz

Frageanzeige im Spielmodus Quiz:



Abbildung 40: Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Quiz
Quelle: Eigene Darstellung, erstellt mit Figma

Frageanzeige im Spielmodus Quiz mobile Teilnehmendenansicht:



Abbildung 41: Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Quiz in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma

Antwortanzeige im Spielmodus Quiz:



Abbildung 42: Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Quiz angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma

Antwortanzeige im Spielmodus Quiz mobile Teilnehmendenansicht:

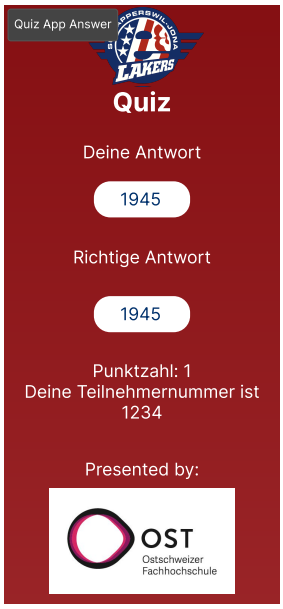


Abbildung 43: Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Quiz in der mobilen Teilnehmendenansicht angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma

A.3.3 Entwurfsansichten des Spielmodus Kreuzworträtsel

Frageanzeige im Spielmodus Kreuzworträtsel:



Abbildung 44: Entwurfsansicht des Anmeldebildschirms für den Spielmodus Kreuzworträtsel
Quelle: Eigene Darstellung, erstellt mit Figma

Frageanzeige im Spielmodus Kreuzworträtsel mobile Teilnehmendenansicht:



Abbildung 45: Entwurfsansicht des Anmeldebildschirms für den Spielmodus Kreuzworträtsel in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma

Antwortanzeige im Spielmodus Kreuzworträtsel:

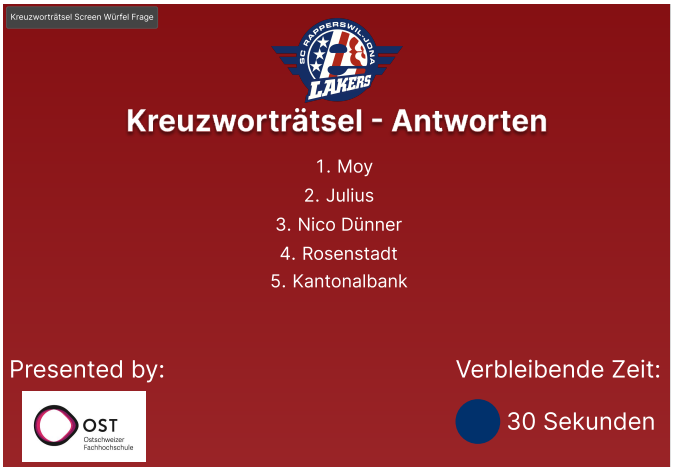


Abbildung 46: Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Kreuzworträtsel angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma

Antwortanzeige im Spielmodus Kreuzworträtsel mobile Teilnehmenden-ansicht:



Abbildung 47: Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Kreuzworträtsel in der mobilen Teilnehmendenansicht angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma

A.3.4 Entwurfsansichten des Spielmodus Bilderrätsel

Frageanzeige im Spielmodus Bilderrätsel:



Abbildung 48: Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Bilderrätsel
Quelle: Eigene Darstellung, erstellt mit Figma

Frageanzeige im Spielmodus Bilderrätsel mobile Teilnehmendenansicht:

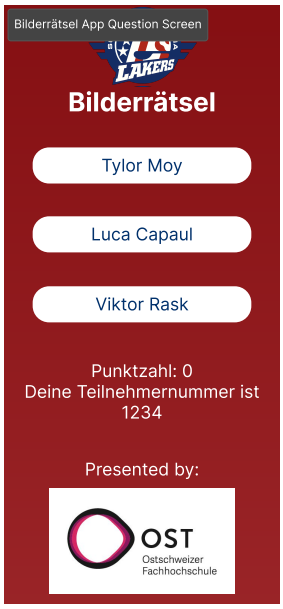


Abbildung 49: Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Bilderrätsel in der mobilen Teilnehmendenansicht
Quelle: Eigene Darstellung, erstellt mit Figma

Antwortanzeige im Spielmodus Bilderrätsel:

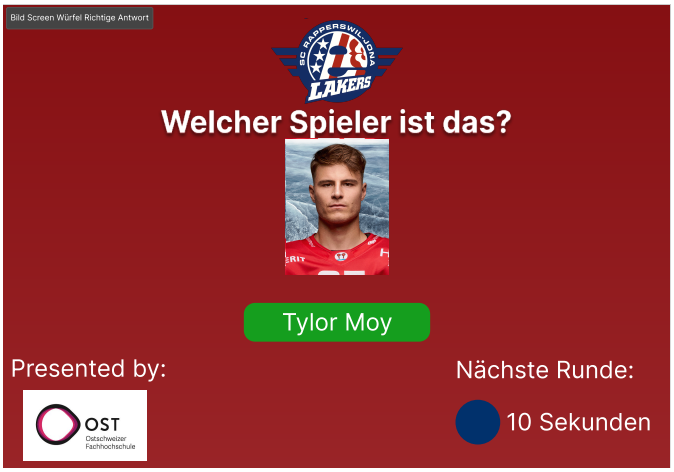


Abbildung 50: Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Bilderrätsel angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma

Antwortanzeige im Spielmodus Bilderrätsel mobile Teilnehmendenansicht:

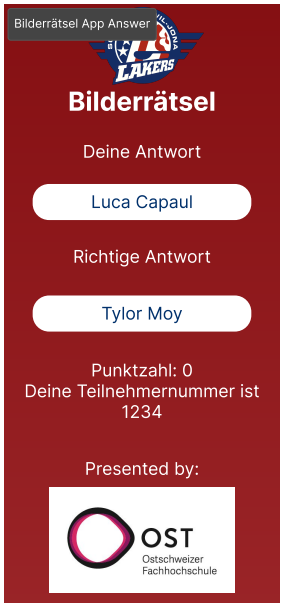


Abbildung 51: Entwurfsansicht des Bildschirms während die Antwort des Spielmodus Bilderrätsel in der mobilen Teilnehmendenansicht angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma

A.3.5 Entwurfsansichten des Administrationsbereichs

Hauptansicht des Administrationsbereichs:

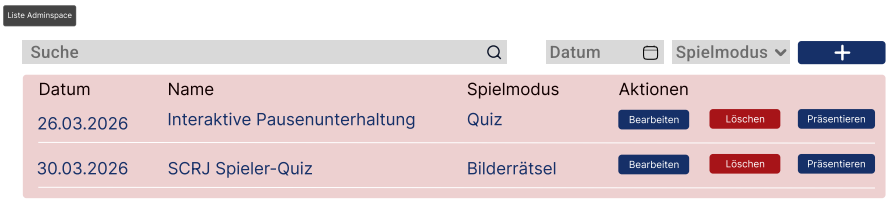


Abbildung 52: Entwurfsansicht der Hauptansicht des Administrationsbereichs
Quelle: Eigene Darstellung, erstellt mit Figma

Spiel im Administrationsbereich mit dem Spielmodus Quiz erstellen:

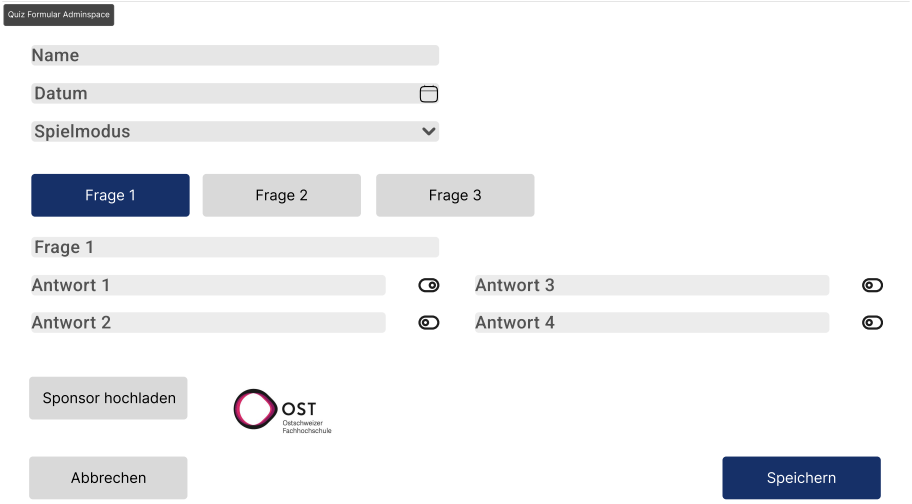


Abbildung 53: Entwurfsansicht zum Erstellen eines Spiels mit dem Spielmodus Quiz im Administrationsbereich
Quelle: Eigene Darstellung, erstellt mit Figma

Spiel im Administrationsbereich mit dem Spielmodus Kreuzworträtsel erstellen:

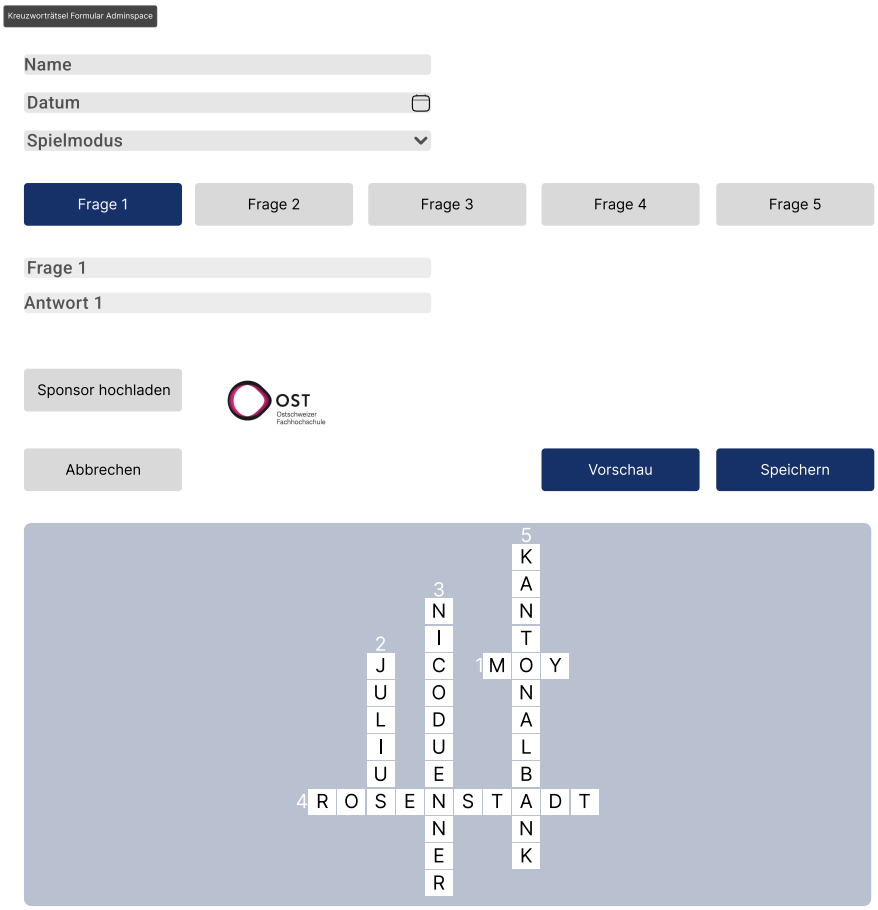


Abbildung 54: Entwurfsansicht zum Erstellen eines Spiels mit dem Spielmodus Kreuzworträtsel im Administrationsbereich
Quelle: Eigene Darstellung, erstellt mit Figma

Spiel im Administrationsbereich mit dem Spielmodus Bilderrätsel erstellen:

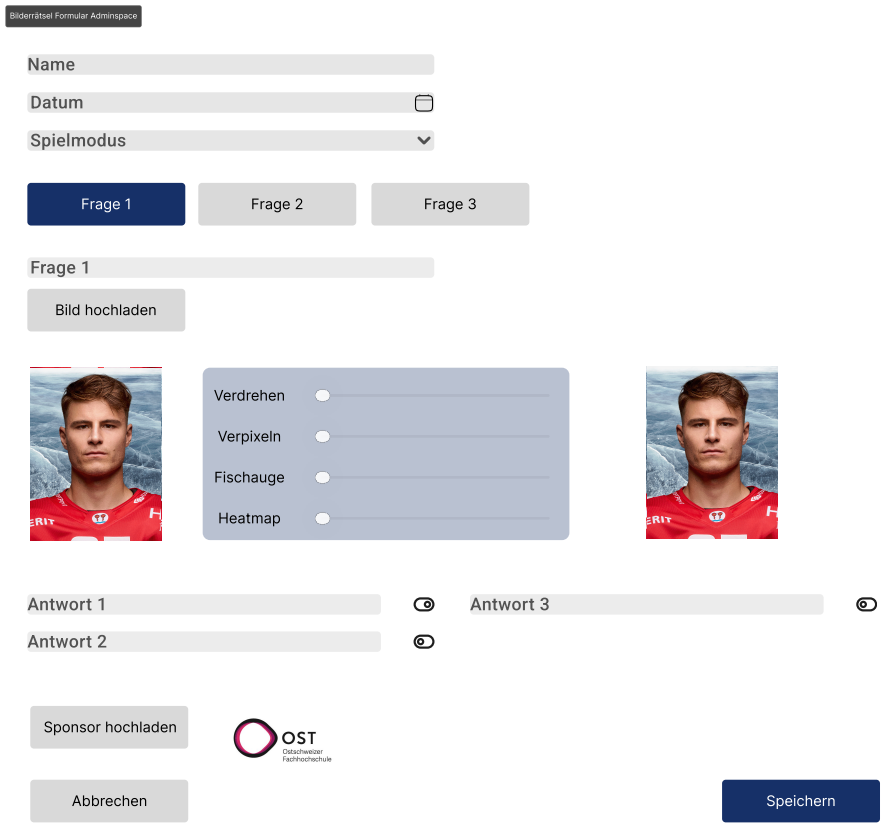


Abbildung 55: Entwurfsansicht zum Erstellen eines Spiels mit dem Spielmodus Bilderrätsel im Administrationsbereich
Quelle: Eigene Darstellung, erstellt mit Figma

A.3.6 Test im Stadion

Foto vom Test der Entwurfsansichten im Stadion:



Abbildung 56: Foto vom Test der Entwurfsansichten im Stadion
Quelle: Eigene Aufnahme



Abbildung 57: Foto vom Test der Entwurfsansichten im Stadion
Quelle: Eigene Aufnahme

A.3.7 Entwurfsansichten Version 2

Überarbeitete Anmeldephase:



Abbildung 58: Überarbeitete Entwurfsansicht des Anmeldebildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma

Überarbeitete Frageanzeige im Spielmodus Kreuzworträtsel:



Abbildung 59: Überarbeitete Entwurfsansicht des Anmeldebildschirms für den Spielmodus Kreuzworträtsel
Quelle: Eigene Darstellung, erstellt mit Figma

Überarbeitete Antwortanzeige im Spielmodus Kreuzworträtsel:



Abbildung 60: Überarbeitete Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Kreuzworträtsel angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma

Überarbeitete Frageanzeige im Spielmodus Bilderrätsel:



Abbildung 61: Überarbeitete Entwurfsansicht des Bildschirms während der Spielphase bei der Anzeige der Frage des Spielmodus Bilderrätsel
Quelle: Eigene Darstellung, erstellt mit Figma

Überarbeitete Antwortanzeige im Spielmodus Bilderrätsel:

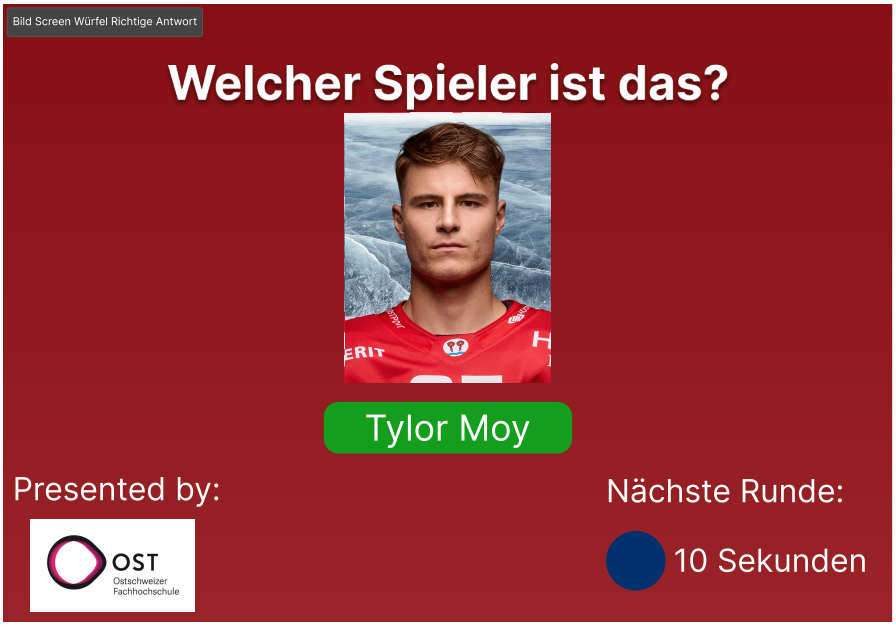


Abbildung 62: Überarbeitete Entwurfsansicht des Bildschirms während die korrekte Antwort des Spielmodus Bilderrätsel angezeigt wird
Quelle: Eigene Darstellung, erstellt mit Figma

Überarbeitete Auswertungsphase:



Abbildung 63: Überarbeitete Entwurfsansicht des Auswertungsbildschirms für alle Spielmodi
Quelle: Eigene Darstellung, erstellt mit Figma

A.4 Screenshots der Anwendung

A.4.1 Administrationsbereich

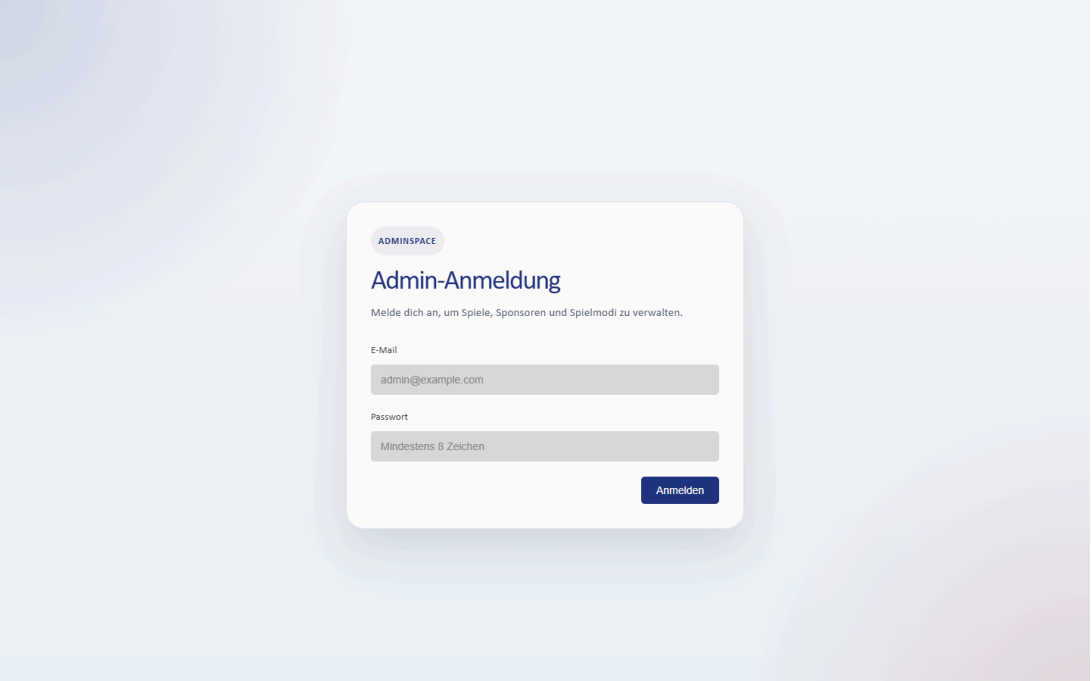


Abbildung 64: Administrationsbereich: Anmeldeansicht für Administrierende.
Quelle: Eigene Aufnahme

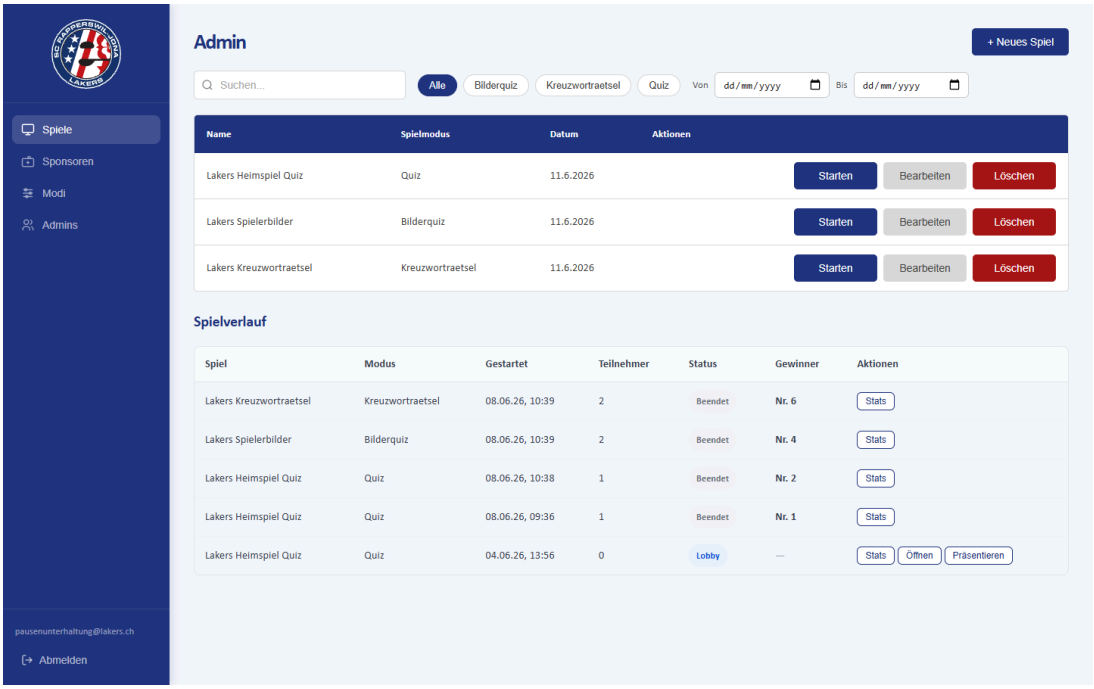


Abbildung 65: Administrationsbereich: Übersicht mit Spielvorlagen, Filtern und Spielverlauf.
Quelle: Eigene Aufnahme

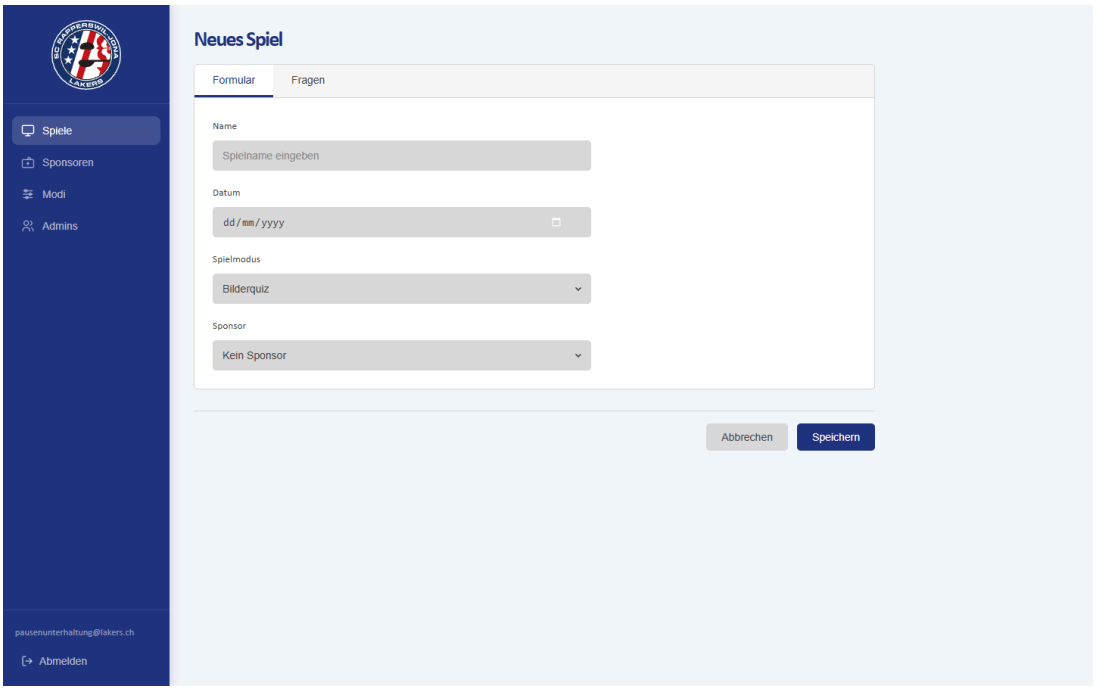


Abbildung 66: Administrationsbereich: Formular zum Erstellen einer neuen Spielvorlage.
Quelle: Eigene Aufnahme

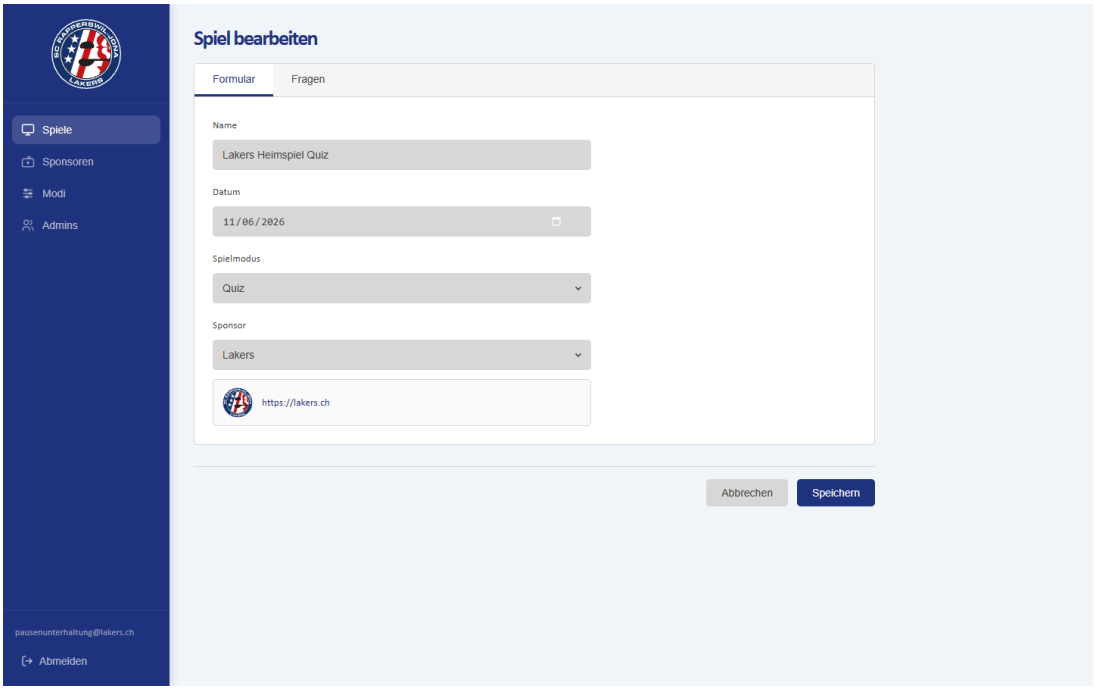


Abbildung 67: Administrationsbereich: Bearbeitung einer Quiz-Spielvorlage.
Quelle: Eigene Aufnahme

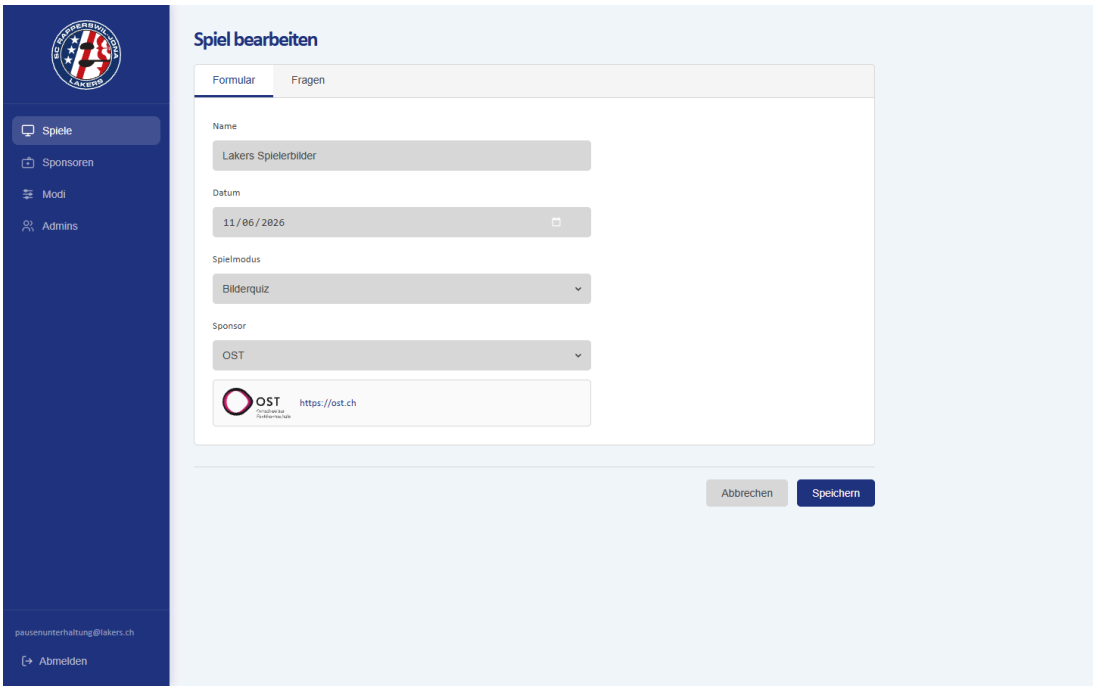


Abbildung 68: Administrationsbereich: Bearbeitung einer Bilderquiz-Spielvorlage.
Quelle: Eigene Aufnahme

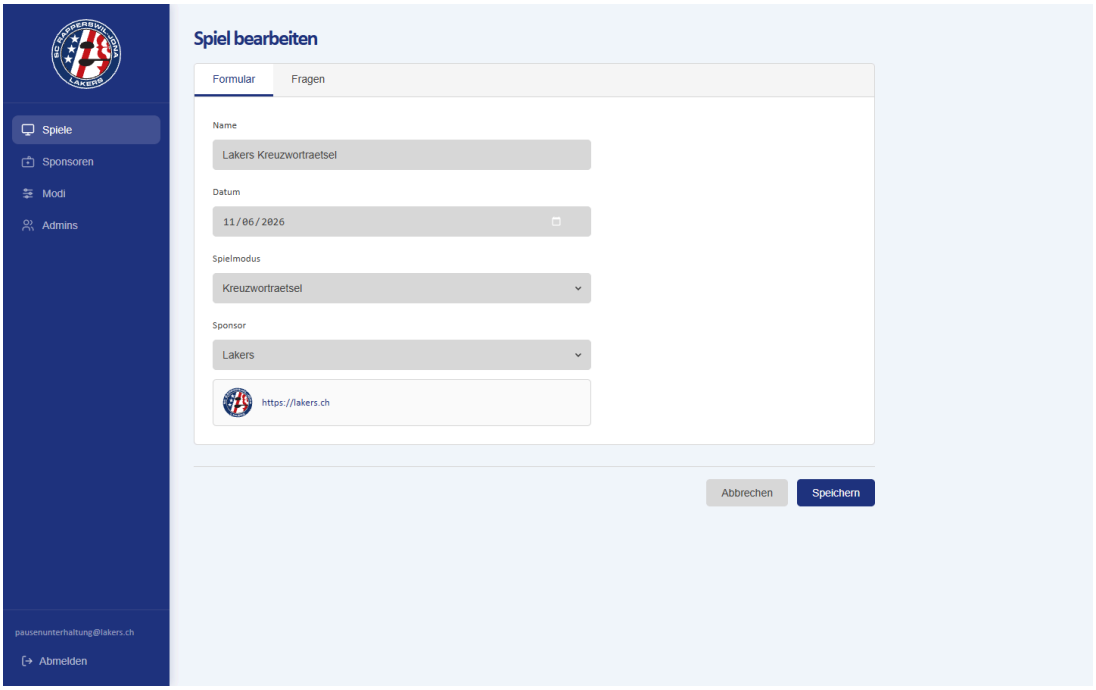


Abbildung 69: Administrationsbereich: Bearbeitung einer Kreuzworträtsel-Spielvorlage.
Quelle: Eigene Aufnahme

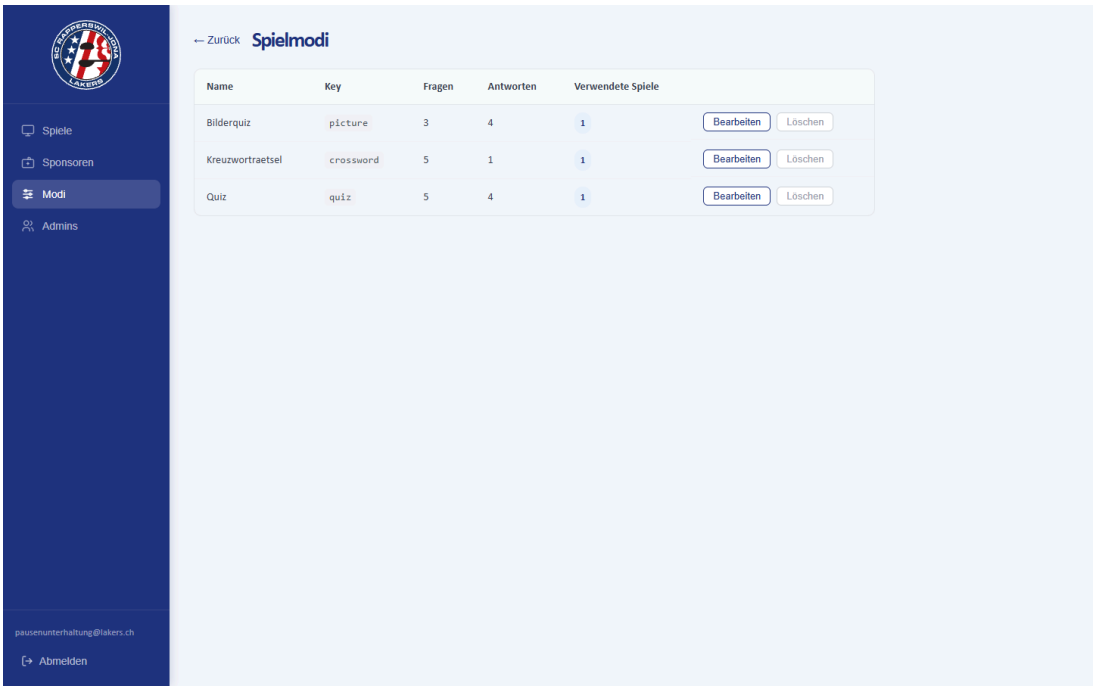


Abbildung 70: Administrationsbereich: Verwaltung der verfügbaren Spielmodi.
Quelle: Eigene Aufnahme

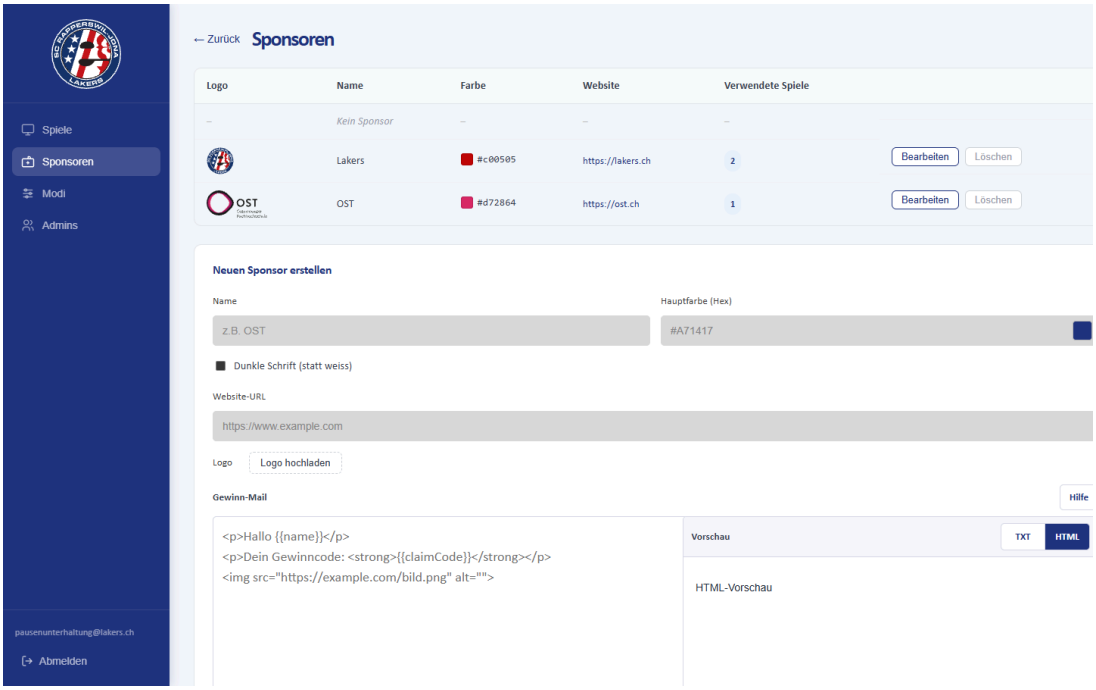


Abbildung 71: Administrationsbereich: Verwaltung der Sponsoren und Sponsoren-daten.
Quelle: Eigene Aufnahme

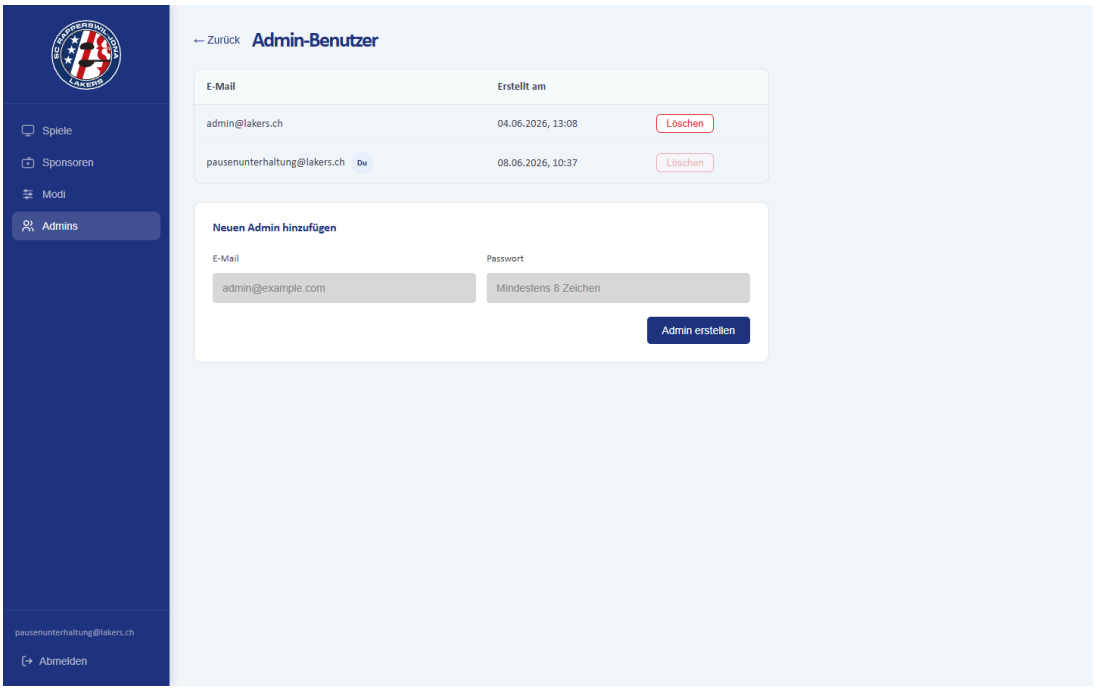


Abbildung 72: Administrationsbereich: Verwaltung der Administrationskonten.
Quelle: Eigene Aufnahme

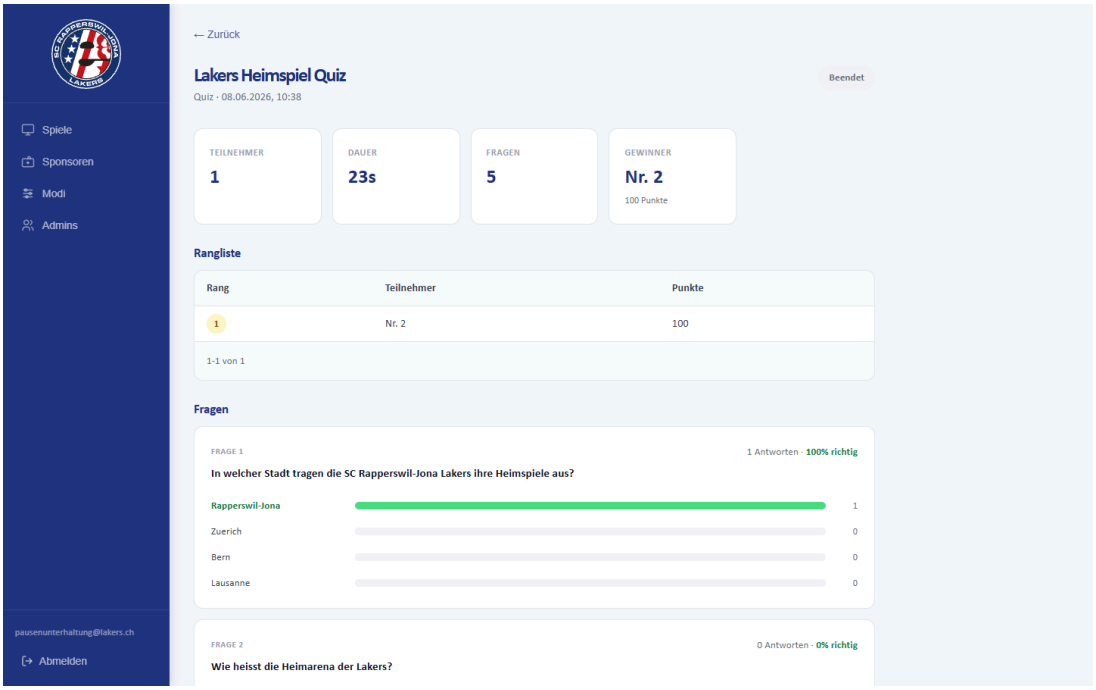


Abbildung 73: Administrationsbereich: Statistikansicht eines abgeschlossenen Quiz-Durchlaufs.
Quelle: Eigene Aufnahme

A.4.2 Präsentationsansicht Quiz



Abbildung 74: Präsentationsansicht Quiz: Wartezustand mit QR-Code und Beitrittscode.
Quelle: Eigene Aufnahme



Abbildung 75: Präsentationsansicht Quiz: Anzeige einer Frage mit Antwortoptionen.
Quelle: Eigene Aufnahme

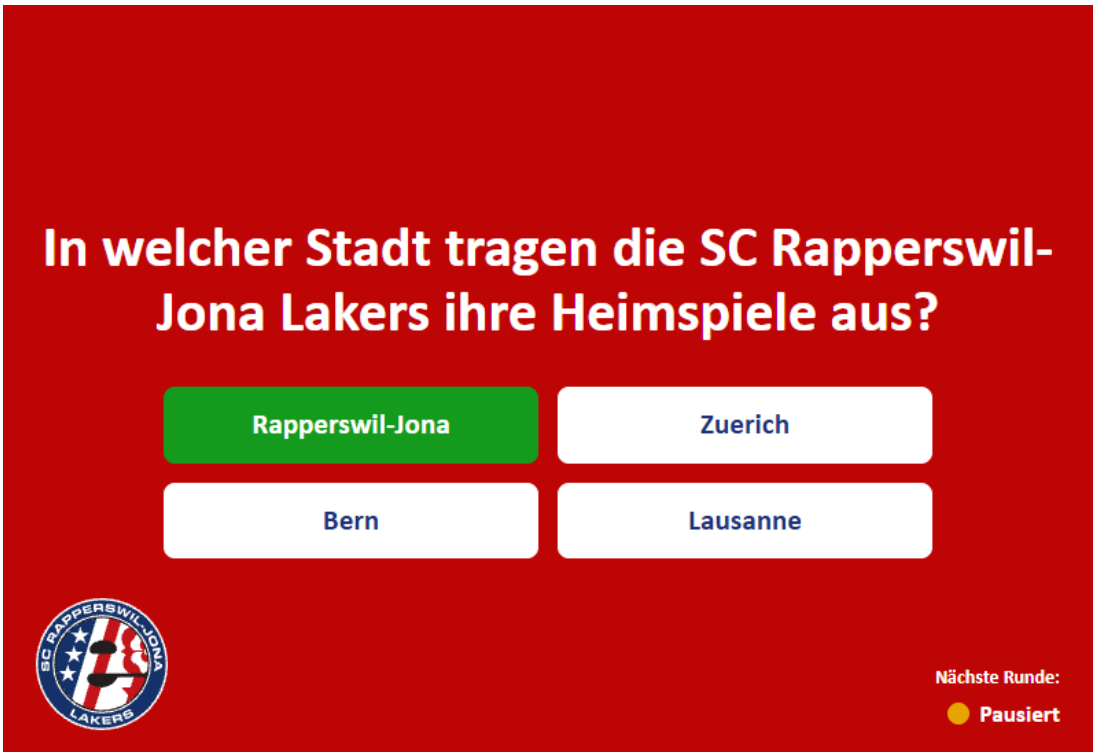


Abbildung 76: Präsentationsansicht Quiz: Auflösung der richtigen Antwort.
Quelle: Eigene Aufnahme



Abbildung 77: Präsentationsansicht Quiz: Abschlussansicht nach Spielende.
Quelle: Eigene Aufnahme

A.4.3 Präsentationsansicht Bilderquiz

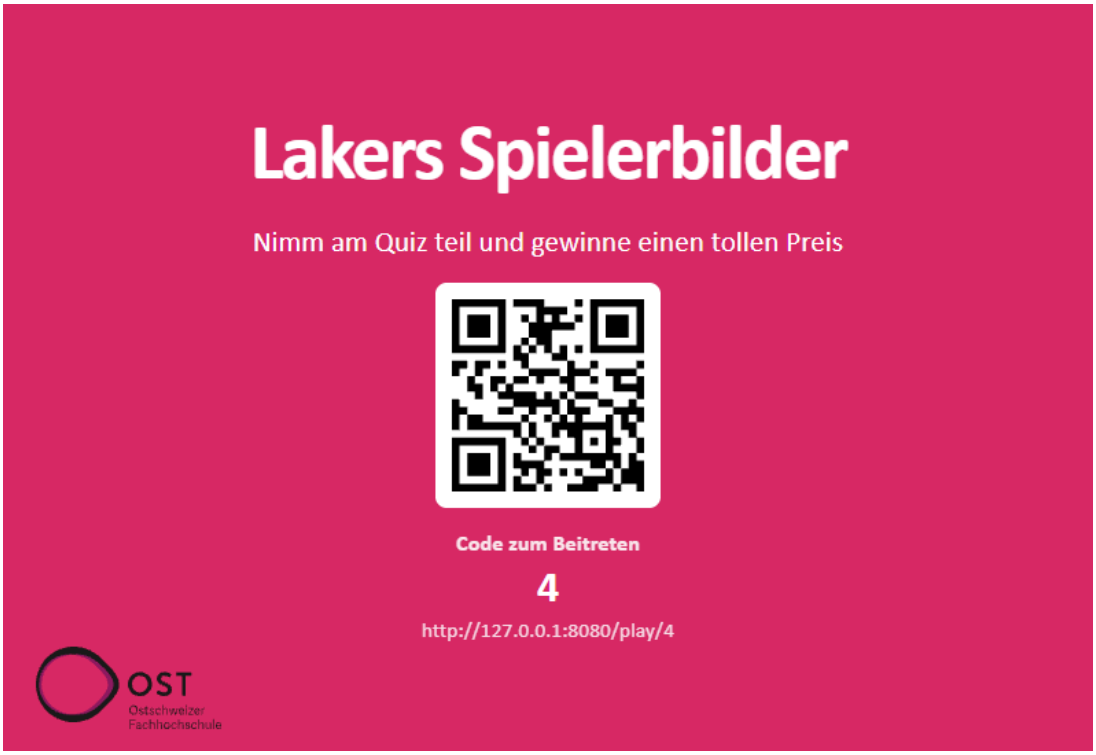


Abbildung 78: Präsentationsansicht Bilderquiz: Wartezustand mit QR-Code und Beitrittscode.
Quelle: Eigene Aufnahme

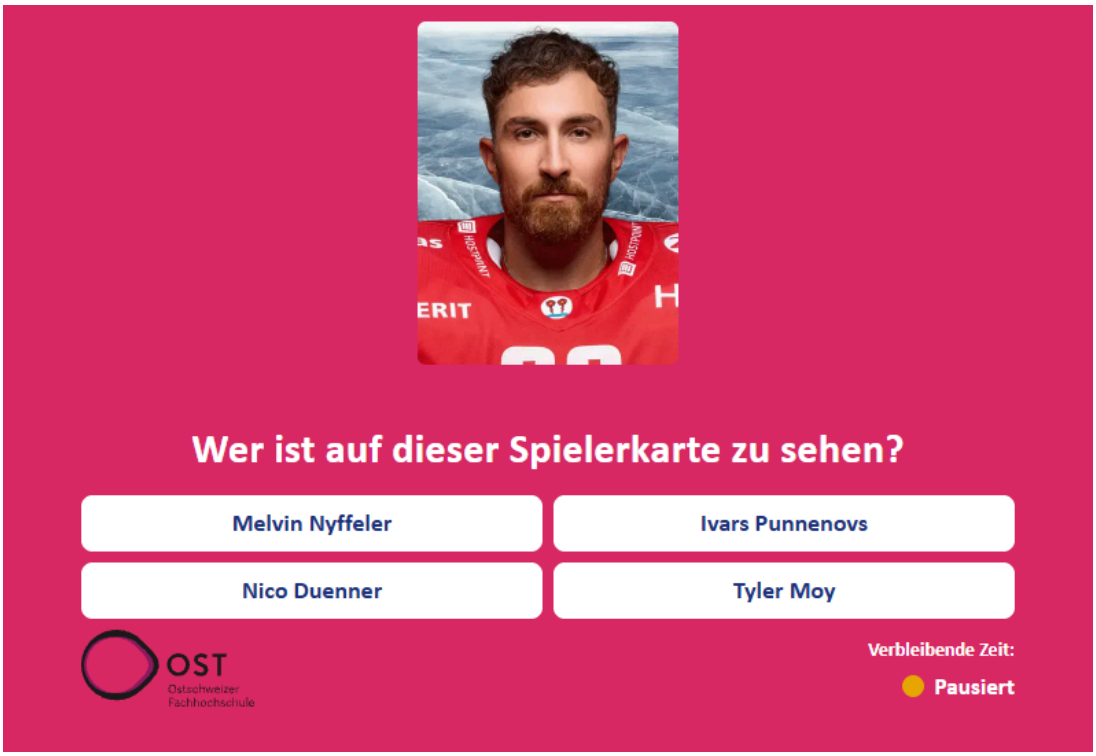


Abbildung 79: Präsentationsansicht Bilderquiz: Anzeige einer Bildfrage.
Quelle: Eigene Aufnahme

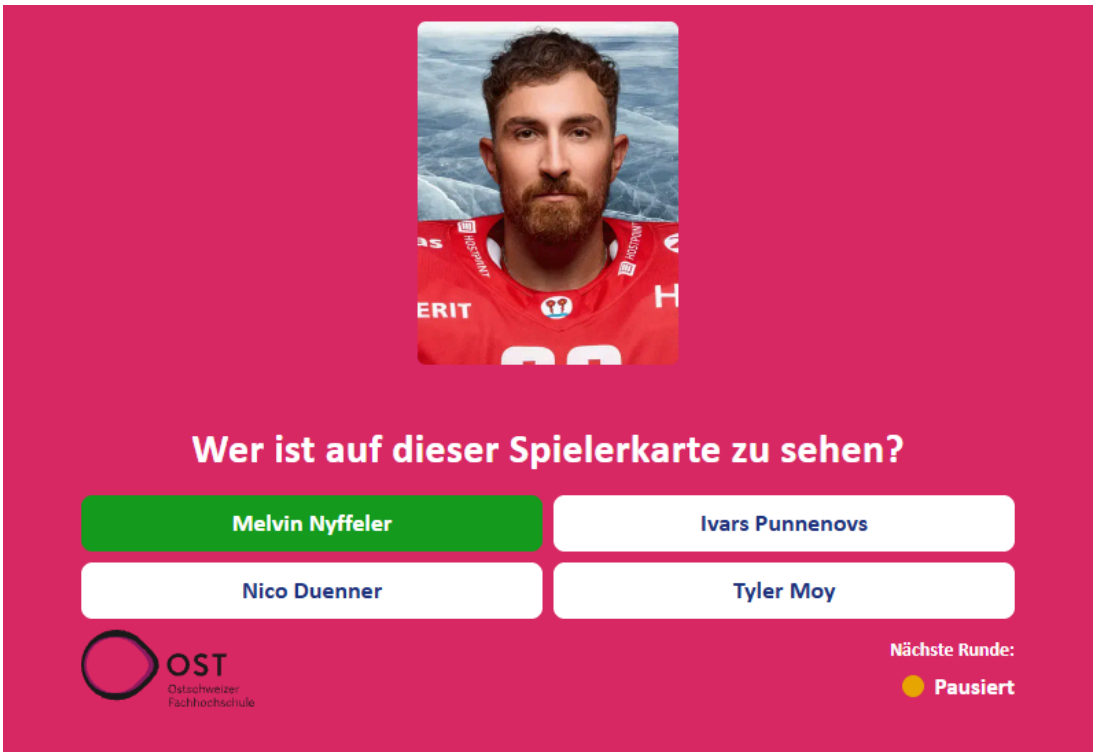


Abbildung 80: Präsentationsansicht Bilderquiz: Auflösung der Bildfrage.
Quelle: Eigene Aufnahme



Abbildung 81: Präsentationsansicht Bilderquiz: Abschlussansicht nach Spielende.
Quelle: Eigene Aufnahme

A.4.4 Präsentationsansicht Kreuzworträtsel



Abbildung 82: Präsentationsansicht Kreuzworträtsel: Wartezustand mit QR-Code und Beitrittscode.
Quelle: Eigene Aufnahme

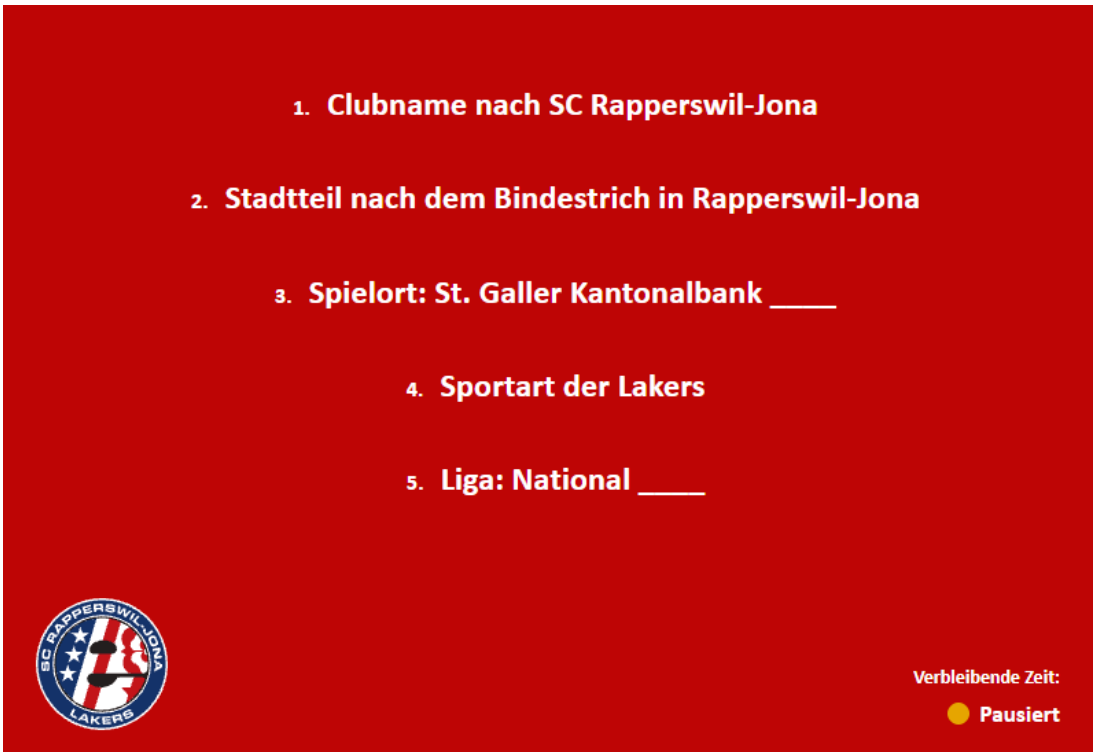


Abbildung 83: Präsentationsansicht Kreuzworträtsel: Anzeige der Hinweise im Fragemodus.
Quelle: Eigene Aufnahme

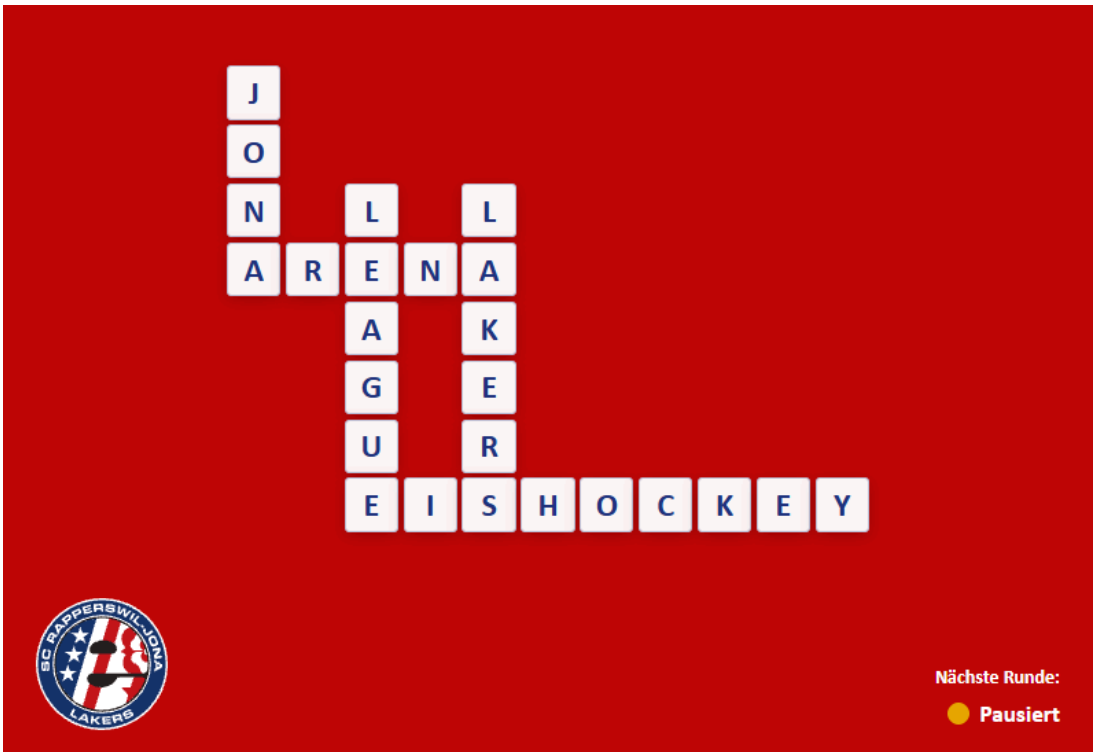


Abbildung 84: Präsentationsansicht Kreuzworträtsel: Auflösung der Antworten.
Quelle: Eigene Aufnahme



Abbildung 85: Präsentationsansicht Kreuzworträtsel: Abschlussansicht nach Spielende.
Quelle: Eigene Aufnahme

A.4.5 Präsentationsansicht mit Administrations-Overlay



Abbildung 86: Präsentationsansicht mit Administrations-Overlay: Quiz im Wartezustand mit Beitrittscode.
Quelle: Eigene Aufnahme

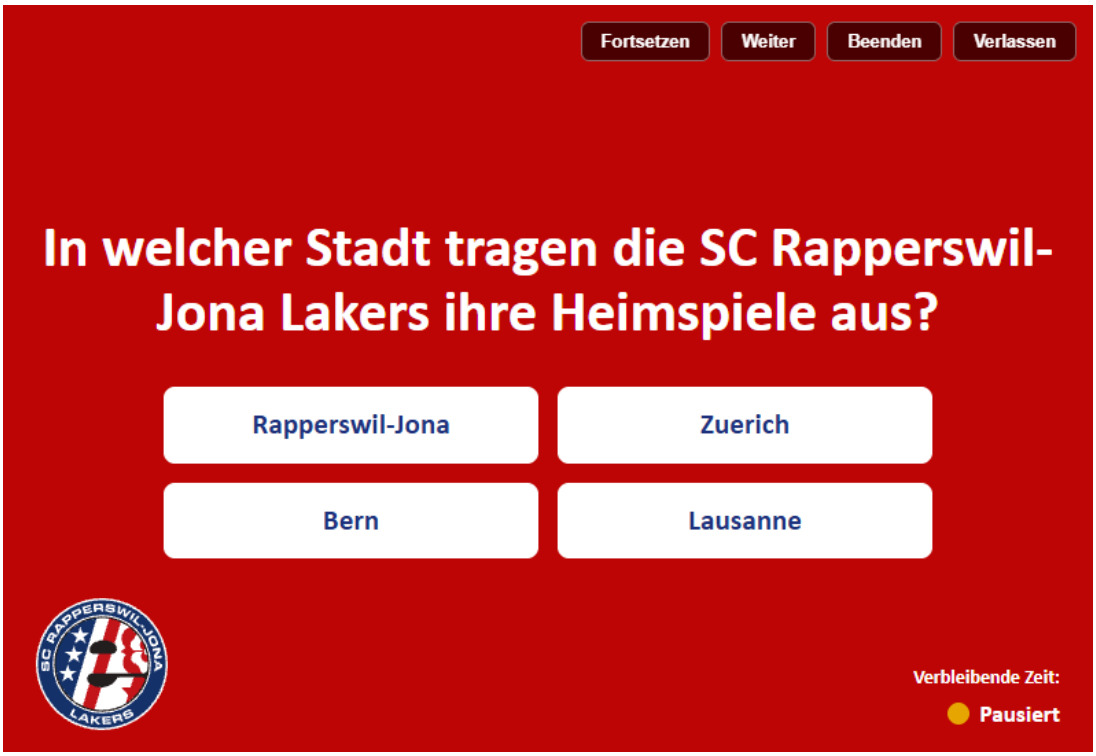


Abbildung 87: Präsentationsansicht mit Administrations-Overlay: Quiz im Fragemodus.
Quelle: Eigene Aufnahme

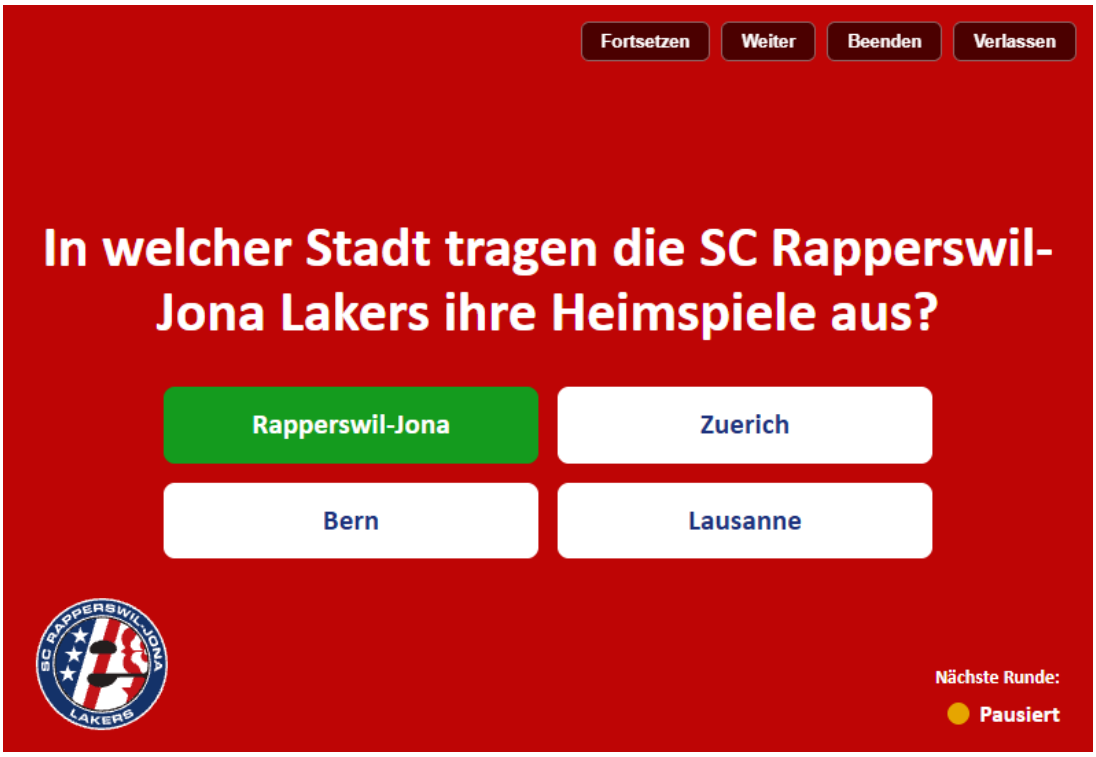


Abbildung 88: Präsentationsansicht mit Administrations-Overlay: Quiz in der Auflösung.
Quelle: Eigene Aufnahme

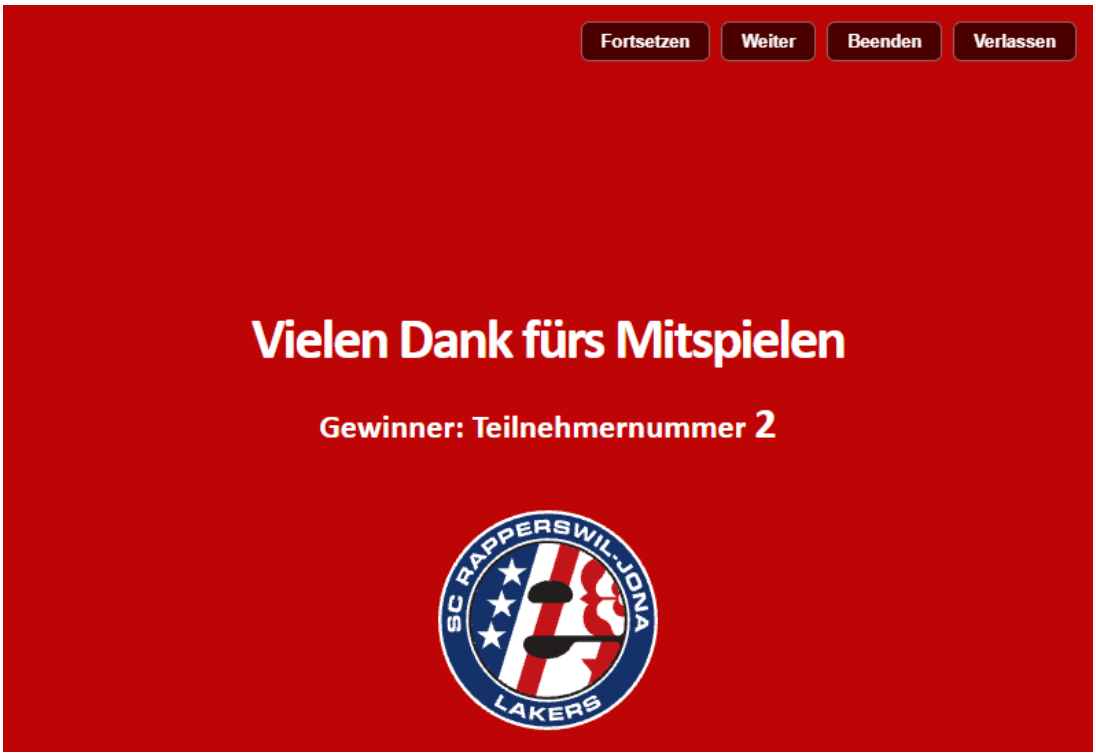


Abbildung 89: Präsentationsansicht mit Administrations-Overlay: Quiz nach Spielende.
Quelle: Eigene Aufnahme

A.4.6 mobile Teilnehmendenansicht Quiz

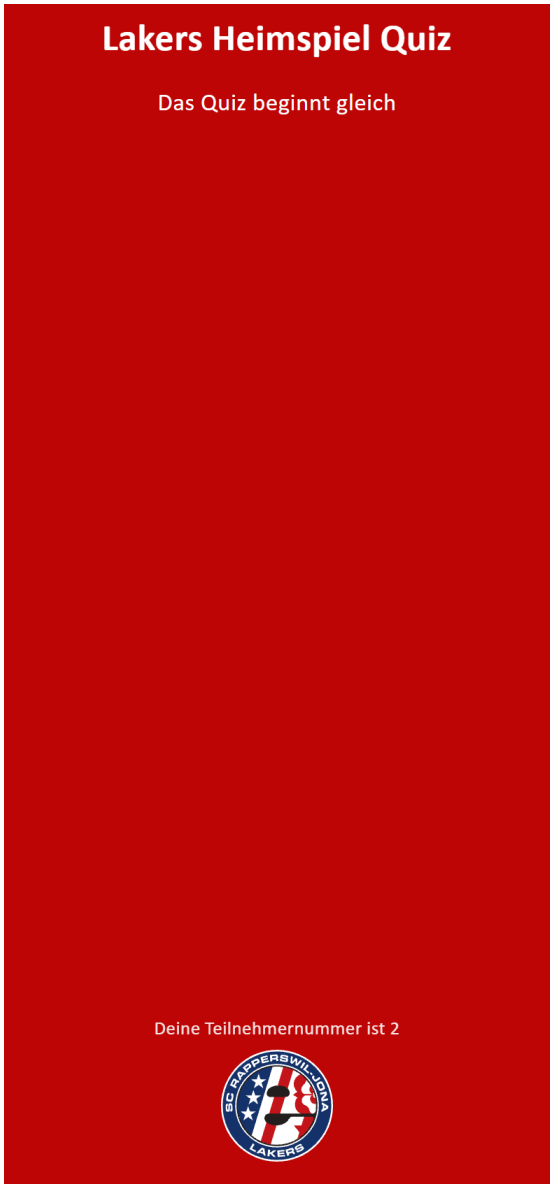


Abbildung 90: mobile Teilnehmendenansicht Quiz: Wartezustand nach dem Beitritt.
Quelle: Eigene Aufnahme

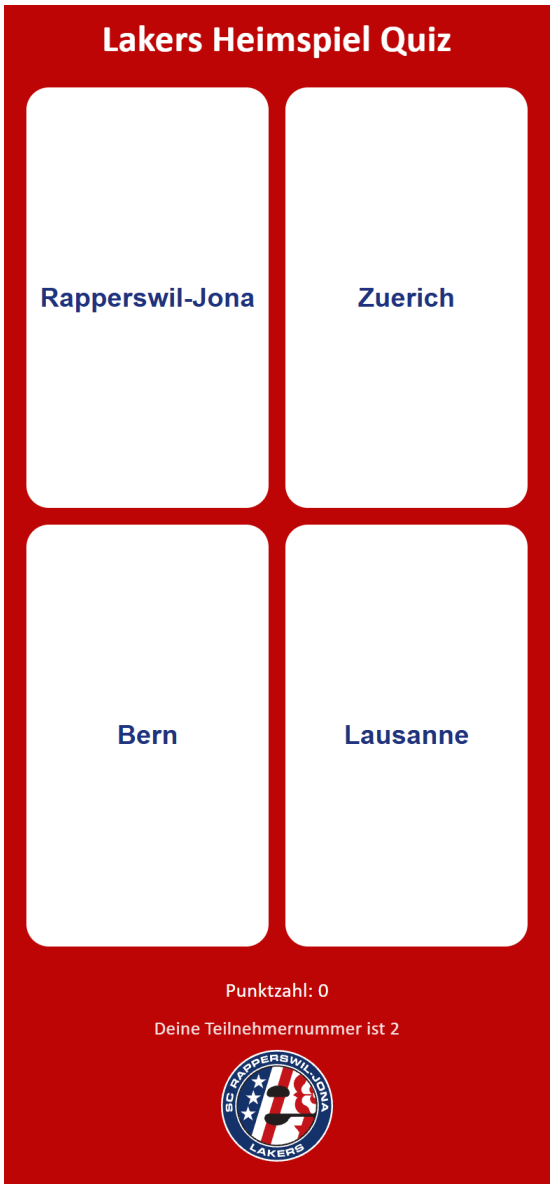


Abbildung 91: mobile Teilnehmendenansicht Quiz: Auswahl einer Antwortoption. Quelle: Eigene Aufnahme

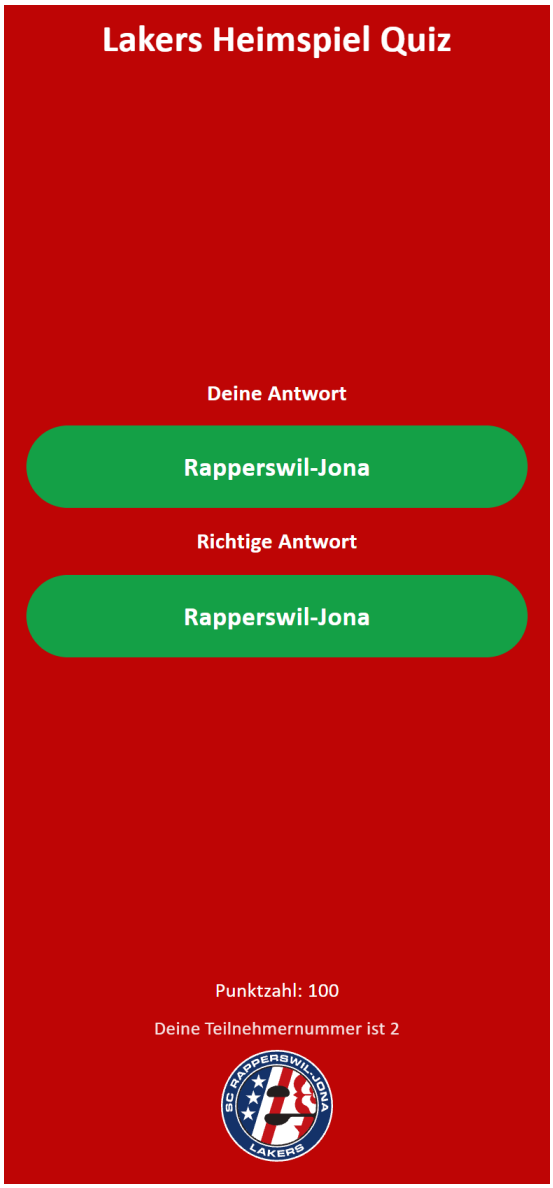


Abbildung 92: mobile Teilnehmendenansicht Quiz: Rückmeldung zur eigenen Antwort.
Quelle: Eigene Aufnahme

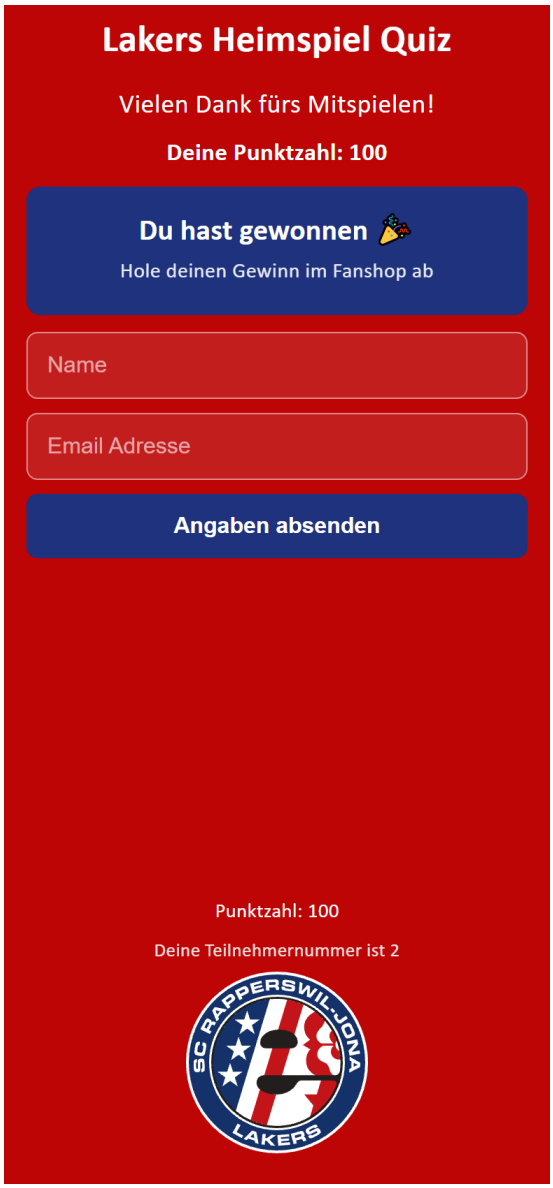


Abbildung 93: mobile Teilnehmendenansicht Quiz: Ergebnisansicht nach Spielende mit eindeutiger Gewinn- oder Nicht-Gewinn-Rückmeldung.
Quelle: Eigene Aufnahme

A.4.7 mobile Teilnehmendenansicht Bilderquiz



Abbildung 94: mobile Teilnehmendenansicht Bilderquiz: Wartezustand nach dem Beitritt.
Quelle: Eigene Aufnahme

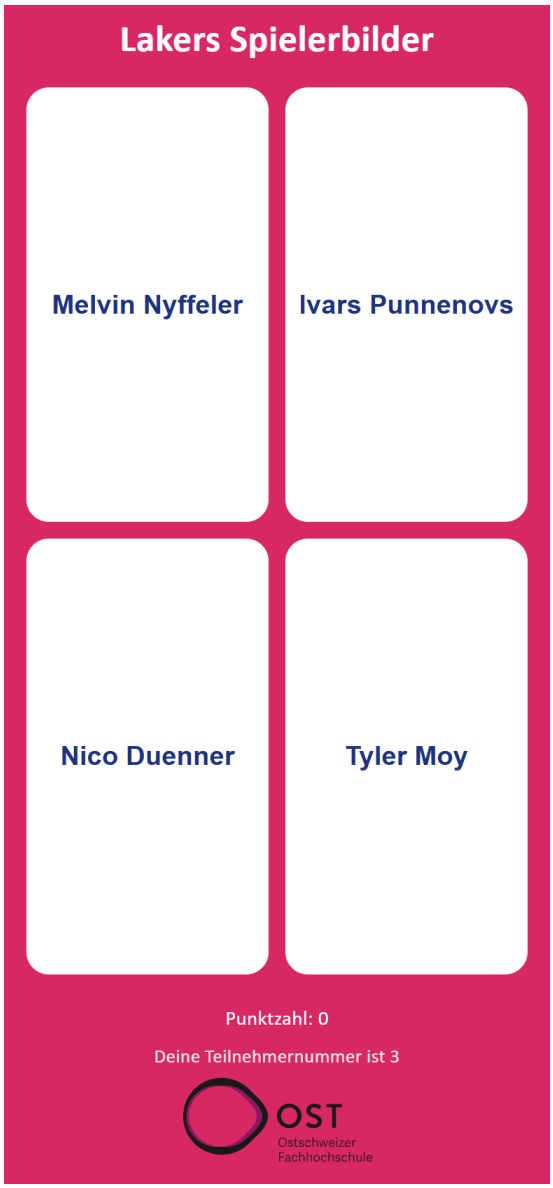


Abbildung 95: mobile Teilnehmendenansicht Bilderquiz: Auswahl einer Antwortoption zur Bildfrage.
Quelle: Eigene Aufnahme

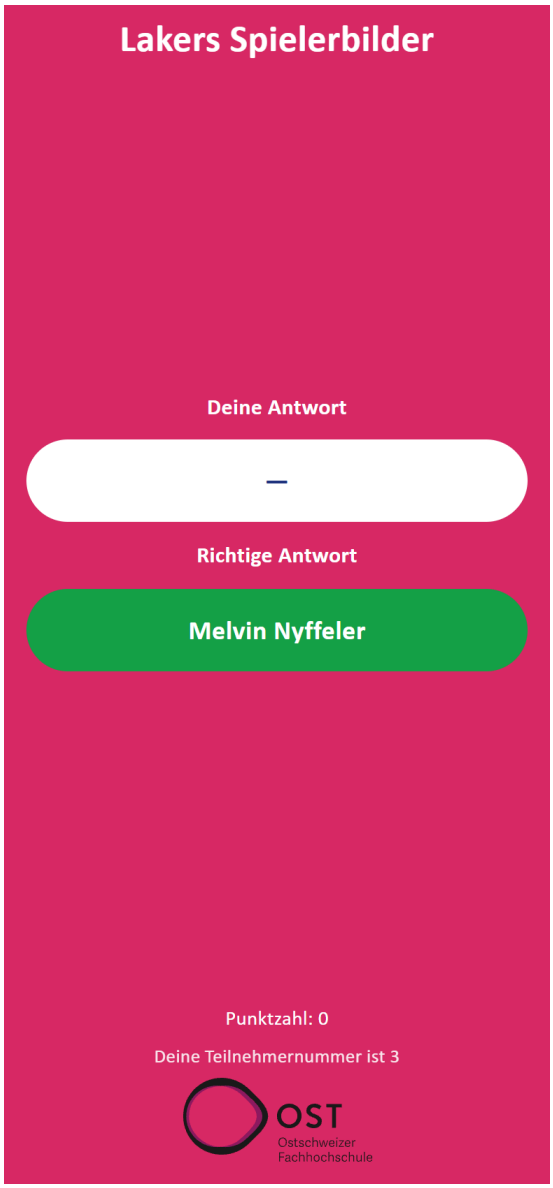


Abbildung 96: mobile Teilnehmendenansicht Bilderquiz: Rückmeldung zur eigenen Antwort.
Quelle: Eigene Aufnahme



Abbildung 97: mobile Teilnehmendenansicht Bilderquiz: Ergebnisansicht nach Spielende mit eindeutiger Gewinn- oder Nicht-Gewinn-Rückmeldung.
Quelle: Eigene Aufnahme

A.4.8 mobile Teilnehmendenansicht Kreuzworträtsel

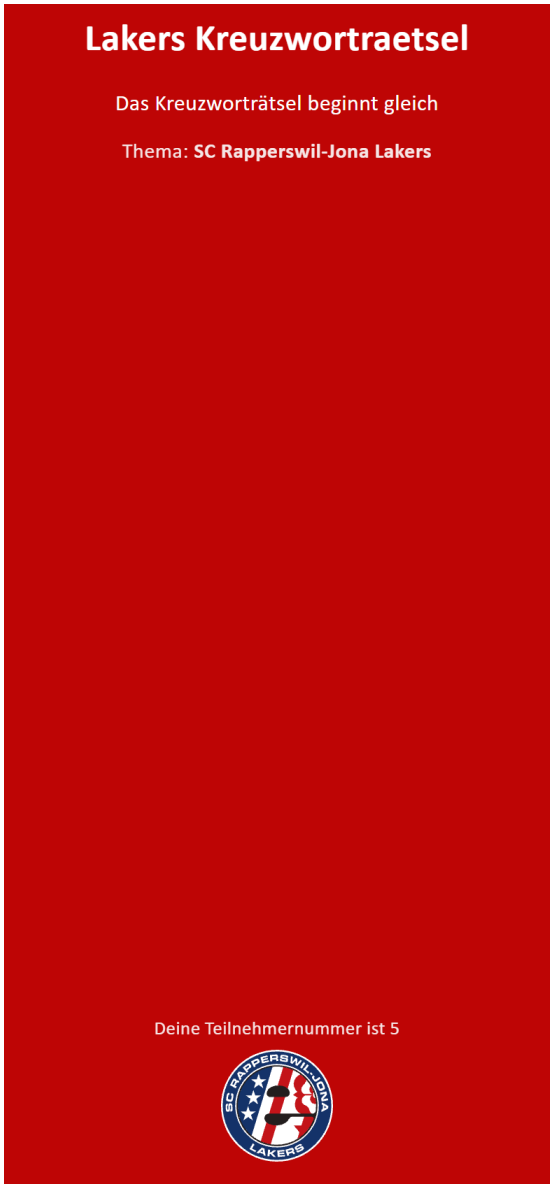


Abbildung 98: mobile Teilnehmendenansicht Kreuzworträtsel: Wartezustand nach dem Beitritt.
Quelle: Eigene Aufnahme

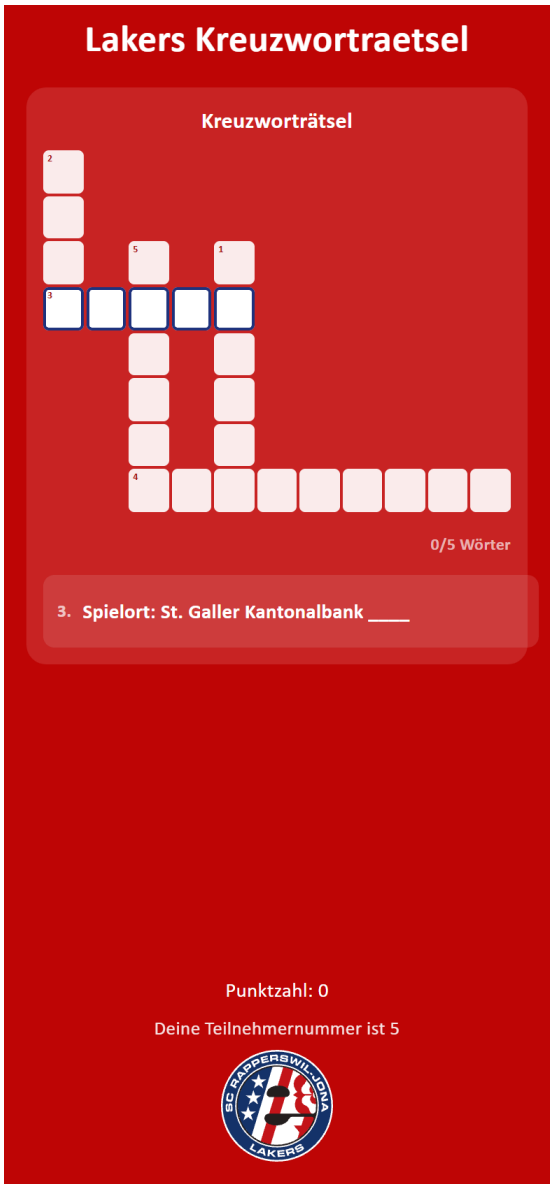


Abbildung 99: mobile Teilnehmendenansicht Kreuzworträtsel: Eingabeansicht für die Wörter.
Quelle: Eigene Aufnahme

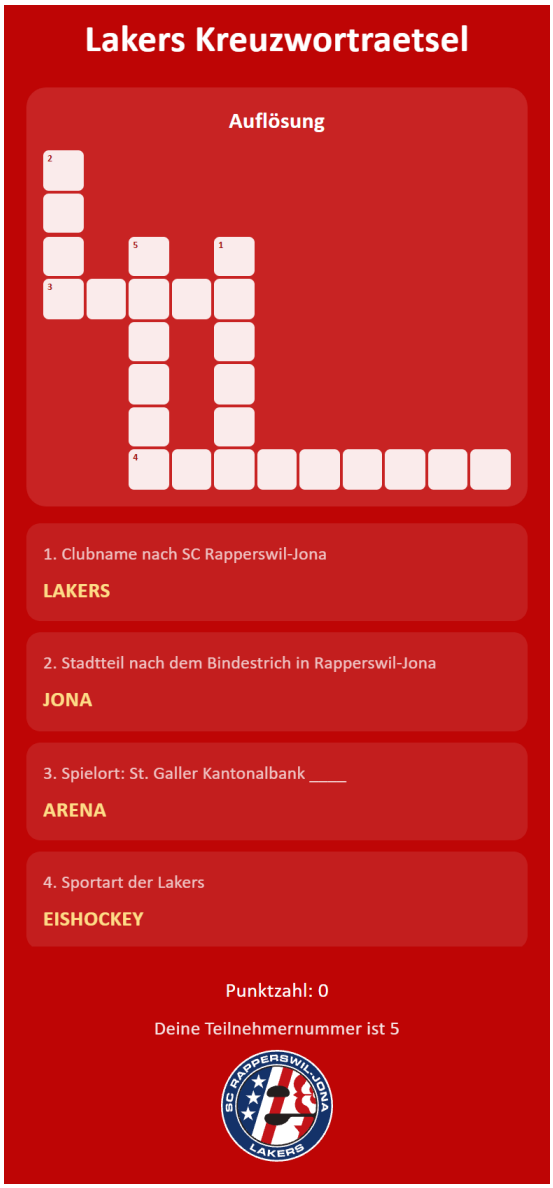


Abbildung 100: mobile Teilnehmendenansicht Kreuzworträtsel: Statische Auflösung der eingegebenen Antworten.
Quelle: Eigene Aufnahme

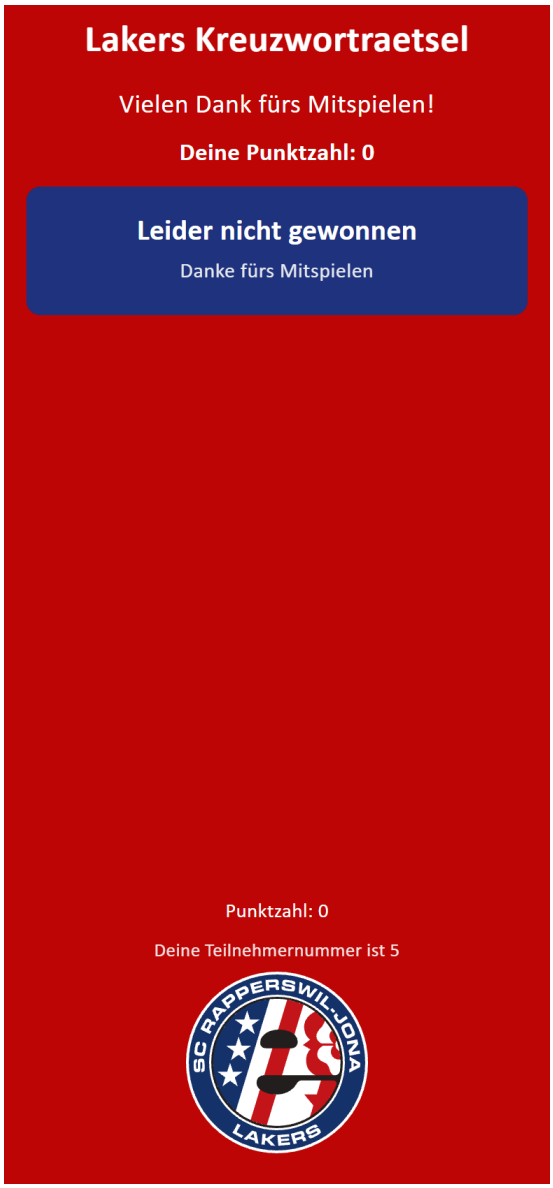


Abbildung 101: mobile Teilnehmendenansicht Kreuzworträtsel: Ergebnisansicht nach Spielende mit eindeutiger Gewinn- oder Nicht-Gewinn-Rückmeldung. Quelle: Eigene Aufnahme

A.5 Publikumszahlenauswertung

Die folgenden Aufnahmen dienten als Grundlage für die Publikumszahlenauswertung in der Analyse.

A.5.1 Spiel 1: Lakersstadion, 27.02.2026

An diesem Spieltag wurden insgesamt 5515 Tickets verkauft.



Abbildung 102: Aufnahme Besuchende 27.02.2026 im Lakersstadion Sektor B mit 290 gezählten Personen nach einer Minute Pause
Quelle: Eigene Aufnahme



Abbildung 103: Aufnahme Besuchende 27.02.2026 im Lakersstadion Sektor C mit 500 gezählten Personen nach einer Minute Pause
Quelle: Eigene Aufnahme



Abbildung 104: Aufnahme Besuchende 27.02.2026 im Lakersstadion Sektor B mit 231 gezählten Personen nach drei Minuten Pause
Quelle: Eigene Aufnahme



Abbildung 105: Aufnahme Besuchende 27.02.2026 im Lakersstadion Sektor C mit 354 gezählten Personen nach drei Minuten Pause
Quelle: Eigene Aufnahme

A.5.2 Spiel 2: Lakersstadion, 05.03.2026

An diesem Spieltag wurden insgesamt 4995 Tickets verkauft.



Abbildung 106: Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor B mit 307 gezählten Personen nach einer Minute Pause
Quelle: Eigene Aufnahme



Abbildung 107: Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor C mit 630 gezählten Personen nach einer Minute Pause
Quelle: Eigene Aufnahme



Abbildung 108: Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor B mit 190 gezählten Personen nach drei Minuten Pause
Quelle: Eigene Aufnahme



Abbildung 109: Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor C mit 460 gezählten Personen nach drei Minuten Pause
Quelle: Eigene Aufnahme

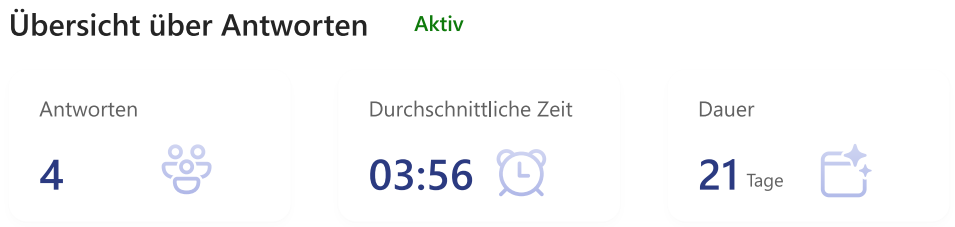


Abbildung 110: Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor B mit 174 gezählten Personen nach vier Minuten Pause
Quelle: Eigene Aufnahme

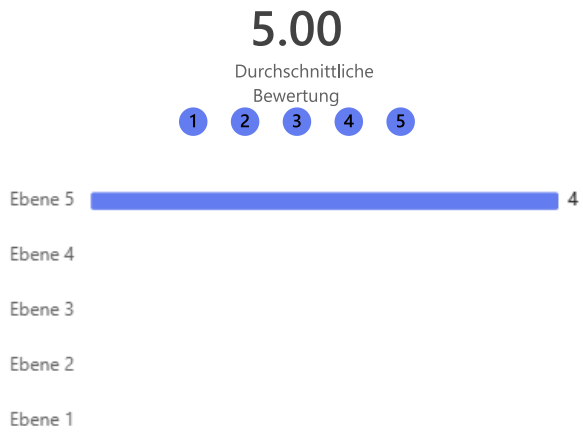


Abbildung 111: Aufnahme Besuchende 05.03.2026 im Lakersstadion Sektor C mit 446 gezählten Personen nach vier Minuten Pause
Quelle: Eigene Aufnahme

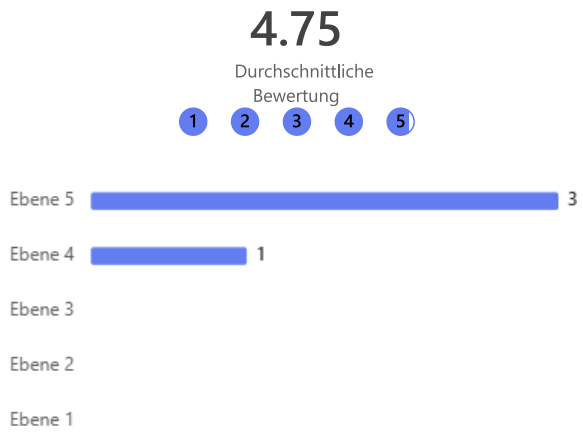
A.6 Umfrageergebnisse Usertest



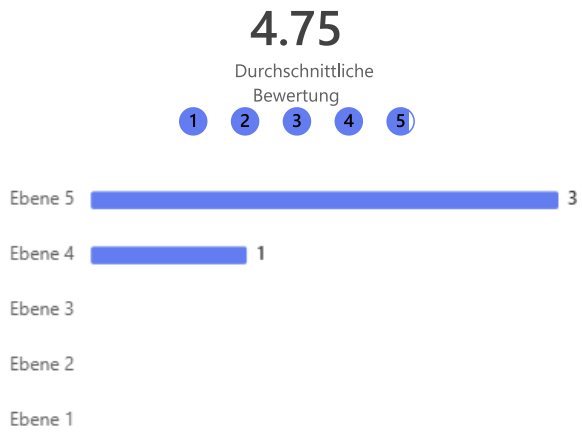
1. Ich konnte der Anwendung ohne Hilfe beitreten.



2. Mir war klar, was ich während des Spiels tun muss.



3. Die Bedienung auf meinem Gerät war verständlich.



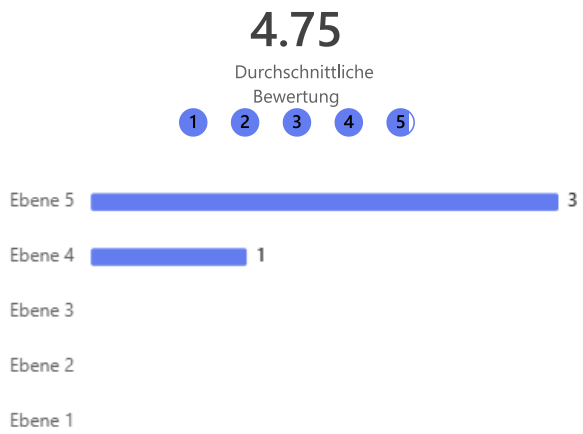
4. Die angezeigten Texte und Rückmeldungen waren verständlich.



5. Die Zeit pro Frage war ausreichend.

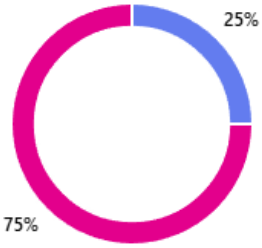


6. Ich würde das Spiel im Stadionkontext wieder spielen.



7. Welcher Modus war am verständlichsten?

Quiz	1
Bilderrätsel	3
Kreutzworträtsel	0



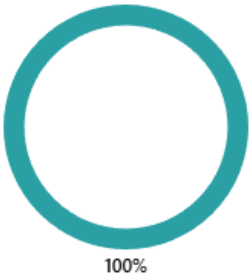
8. Welcher Modus war am unterhaltsamsten?

Quiz	0
Bilderrätsel	2
Kreuzworträtsel	2



9. Bei welchem Modus war die Zeit am knappsten?

Quiz	0
Bilderrätsel	0
Kreutzworträtsel	4



10. Was war für dich unklar?

3
Antworten

Neueste Antworten
"Richtige Antwort: Aufleuchten der Antworten, war ein kur... "
"Nichts"
• • •

11. Gab es einen Moment, in dem du nicht wusstest, was du drücken oder tun musst?

3
Antworten

Neueste Antworten
"Beim Kreuzworträtsel waren zwei Klicks nötig, bis sich die ..."
"Nein"
• • •

12. Was sollte an der Anwendung verbessert werden?

4
Antworten

Neueste Antworten
"Kreuzworträtsel war bei mir gut, bei einem kleinen Handy... "
"Beim kreuzworträtsel werden bei erneuter eingabe buchs... "
"Wenn man nicht gewonnen hat sollte das angezeigt werd... "
• • •

13. Was hat dir besonders gut gefallen?

3
Antworten

Neueste Antworten
"Kreuzworträtsel Live gegeneinander ist noch cool."
"Bedienung ansonsten eigentlich sehr angenehm"
"Einfachheit"

14. Weitere Bemerkungen:

0 übermittelte Antworten

0

Antworten

A.7 Qualitätsstandart

Dieser Abschnitt beschreibt die Quality Measures und Projekt-Guidelines, die für Konsistenz im Team und hohe Qualität in allen Projektbereichen sorgen.

A.7.0.1 Dokumentation

- Die Dokumentsprache ist Deutsch. Fachbegriffe oder übliche Überschriften können in Englisch geschrieben werden.
- Die Dokumentation wird in Typst verfasst.
- Für jedes Hauptkapitel (z.B. 5) wird ein Ordner erstellt und für jedes Unterkapitel (z.B. 5.1) ein eigenes Typst-File erstellt.
- Namenskonvention für Ordner und Files in Typst: Kapitelnummer und Kapitelname in PascalCase (z.B. 04_QualityMeasures/01_WorkingEnvironment.typ)
- Um die Zusammenarbeit und das Mergen in Git zu vereinfachen, sollte jeder Satz in einer neuen Zeile begonnen werden.

A.7.0.2 Code

- Um eine gute Code-Qualität und eine einheitliche Code-Formatierung zu gewährleisten sowie Fehler frühzeitig zu erkennen, werden Linters eingesetzt.
- Der Code wird, wo passend, auf Englisch kommentiert.
- Klassen und Variablennamen sind aussagekräftig, einheitlich benannt und folgen den sprachüblichen Conventions (z.B. PascalCase für Klassen, camelCase für Variablen und Methoden).

A.7.0.3 Definition of Done

Ein Task wird nur dann abgeschlossen, wenn folgende Kriterien erfüllt sind:

- Die oben genannten Code-Guidelines sind umgesetzt.
- Die Pipeline ist erfolgreich durchgelaufen.
- Der Featurebranch wurde in den Mainbranch integriert.
- Alle im YouTrack-Ticket definierten Akzeptanzkriterien sind erfüllt.
- Zeit wurde in YouTrack eingetragen.

A.7.0.4 Version Control

Für die Versionskontrolle des Source Codes und der Dokumentation wird GitLab verwendet, und die Commits und Branches werden auf Englisch verfasst. Genaue Guidelines werden in den folgenden Kapiteln beschrieben.

Branching

Für einzelne Tasks wird jeweils ein eigener Branch erstellt. Hierfür werden die folgenden Naming Conventions verwendet:

Aufbau eines Branchnamens

Der Branchname wird in kebab-case geschrieben und hat folgenden Aufbau:

<Prefix>/<Task_id>-<kurze_Beschreibung>

z.B. „docs/bjs-29-test-concept“

Passender Prefix:

Prefix	Anwendung
feature/	Für die Entwicklung neuer Features
bugfix/	Um Bugs im Code zu beheben, oft in Verbindung mit einem Issue erstellt
hotfix/	Um kritische Bugs in der Produktion zu beheben
release/	Zur Vorbereitung einer neuen Version, typischerweise für letzte Anpassungen und Überprüfungen
docs/	Zum Schreiben, Ändern oder Korrigieren der Dokumentation
ci/	Für Änderungen an der Continuous Integration/Continuous Deployment (CI/CD)-Pipeline

Tabelle 50: Branch Prefixes

Commits

Es gelten folgende Guidelines für alle Commit Messages:

Head = Maximale Länge von 50 Zeichen (Grobe Beschreibung)

Body = Maximale Länge von 72 Zeichen pro Zeile, kann mehrzeilig sein

Aufbau einer Commit-Message

Zeile 1: <Type>(<Task_id>): <Subject> (z.B. „feat(17): Add Admin-View“)

Zeile 2: Leerzeile

Zeile 3: Body des Commits mit einer genaueren Beschreibung (optional)

Passender Type:

Type	Anwendung
build	Änderungen im Zusammenhang mit dem Build-Prozess (z.B. Hinzufügen von npm-Abhängigkeiten oder externen Bibliotheken)
feat	Ein neues Feature
fix	Ein Bugfix
docs	Änderungen an der Dokumentation
ci	Änderungen an der Continuous Integration/Continuous Deployment (CI/CD)-Pipeline
refactor	Code, der weder einen Bug fixt, noch ein Feature hinzufügt (z.B. semantische Änderungen wie das Umbenennen einer Variable oder Funktion)
perf	Code, der die Performance verbessert
test	Hinzufügen neuer Tests oder Änderungen an bestehenden Tests

Tabelle 51: Commit Types

Merge Request

Nach Fertigstellung einer Änderung wird ein Merge Request erstellt. Dieser muss von der anderen Person geprüft und freigegeben werden.

Der Name des Merge Requests entspricht dem Namen des zugehörigen Branches.

A.7.0.5 Review Process

Nachdem ein Merge Request erstellt wurde, muss die Story in YouTrack in die Spalte „Wird geprüft“ gezogen und der anderen Person zugewiesen werden. Wenn das Review positiv verläuft, wird es wieder dem Entwickler zugewiesen und in die Spalte „Mergen“ gezogen. Bei einem negativen Review wird die Story in die Spalte „in Bearbeitung“ anstelle von „Mergen“ gezogen.

A.7.0.6 Automatic Quality Control

Bei Fehlern schlägt die Pipeline fehl und der Merge Request kann nicht gemergt werden, bis die Fehler behoben sind.