

## Studie

Status	In Arbeit
Projektname	xServer
Projektleiter	Maurice Däppen
Auftraggeber	Valve (Steam)
Autoren	Maurice Däppen, Patrick Aeschlimann, Lenny Herren, Mika Hannappel
Verteiler	Maurice Däppen, Patrick Aeschlimann, Lenny Herren, Mika Hannappel, Christian Kissling

### Änderungskontrolle, Prüfung, Genehmigung

Version	Datum	Beschreibung, Bemerkung	Name oder Rolle
1.0.0	21.02.2024	Erstellung	Maurice Däppen

## Definitionen und Abkürzungen

Begriff / Abkürzung	Bedeutung

## Referenzen

Referenz	Titel, Quelle
[1]	
[2]	
[3]	

## Inhaltsverzeichnis

1	Situationsanalyse .....	4
1.1	Ausgangslage.....	4
1.2	Stärken .....	5
1.3	Schwächen.....	5
2	Ziele.....	6
2.1	Rahmenbedingungen .....	7
2.2	Abgrenzung .....	7
3	Liste der Stakeholder .....	8
4	Anforderungen (Initial Product Backlog) .....	8
5	Lösungsvarianten .....	10
5.1	Variantenübersicht.....	10
5.2	Beschreibung der Varianten .....	10
6	Bewertung der Varianten (Tabelle).....	11
7	Lösungsbeschreibung .....	14
8	Projektplanung .....	15

## Abbildungsverzeichnis

## 1 Situationsanalyse

### 1.1 Ausgangslage

Das Projekt zielt darauf ab, eine umfassende und intuitiv bedienbare Online-Plattform zu entwickeln, die es Einzelpersonen und kleinen Gaming-Communities ermöglicht, ihre eigenen privaten Gameserver für eine Vielzahl von unterstützten Spielen, wie Minecraft oder Ark, zu erstellen, anzupassen und effizient zu verwalten. Der Schwerpunkt liegt darauf, den Nutzern eine benutzerfreundliche Oberfläche zu bieten, die es ihnen erlaubt, ohne umfangreiches technisches Wissen Server nach ihren eigenen Vorstellungen zu konfigurieren – von der Auswahl der Hardware-Spezifikationen wie CPU-Kapazität und RAM bis hin zu spezifischen Spieleinstellungen. Darüber hinaus soll die Plattform eine sichere und bequeme Zahlungsabwicklung über gängige Online-Zahlungsmethoden, insbesondere das Twint-Plugin, unterstützen, um den Kaufprozess so reibungslos wie möglich zu gestalten.

Die Problemstellung, die dieses Projekt adressiert, ist vielschichtig. Viele Gaming-Enthusiasten und -Communities stehen vor der Herausforderung, eigene Gameserver aufzusetzen, da dies oft ein hohes Mass an technischem Know-how erfordert. Sie müssen sich mit komplexen Themen wie Serverhardwareonfiguration, Netzwerkeinstellungen, Sicherheitsprotokollen und der Installation sowie Konfiguration der Spielsoftware auseinandersetzen. Diese technischen Barrieren können abschreckend wirken und verhindern, dass Spieler ihre idealen Spielumgebungen selbst gestalten können. Das Projekt zielt darauf ab, diese Hindernisse zu beseitigen, indem es eine zugängliche, nutzerorientierte Lösung anbietet, die die technische Komplexität verbirgt und den Nutzern die Freiheit gibt, sich auf das Spielerlebnis zu konzentrieren.

Der Nutzen des Projekts erstreckt sich auf mehrere Ebenen. Erstens ermöglicht es Spielern, ihre Spielumgebungen nach ihren Wünschen zu personalisieren, was zu einem verbesserten und individuelleren Spielerlebnis führt. Zweitens senkt es die Einstiegshürden für das Hosting von Gameservern, indem es technische Prozesse vereinfacht und automatisiert, was die Plattform für eine breitere Zielgruppe zugänglich macht. Darüber hinaus bietet es eine skalierbare Lösung, die sich den wachsenden Anforderungen der Nutzer anpassen kann, von kleinen Gruppen, die einen Server für Freunde hosten wollen, bis hin zu grösseren Communities, die eine leistungsfähigere Infrastruktur benötigen.

Bislang wurden in der Vorbereitungsphase des Projekts grundlegende konzeptionelle Überlegungen angestellt, insbesondere im Hinblick auf die Monetarisierungsstrategie und die Auswahl der Hosting-Infrastruktur. Es wurde ein erstes Geschäftsmodell skizziert, das auf der Bereitstellung von Server-Hosting-Diensten gegen Bezahlung basiert, wobei ein besonderer Fokus auf der Integration sicherer und benutzerfreundlicher Zahlungsmethoden liegt. Die Entscheidung, Ressourcen auf AWS oder AZURE zu nutzen, unterstreicht das Bestreben, eine hochverfügbare und skalierbare Lösung anzubieten. Technische Details, wie die Verwendung von Terraform zur Automatisierung der Infrastrukturbereitstellung und CloudInit-Skripte für die Serverkonfiguration, wurden als potenzielle Methoden zur Vereinfachung des Bereitstellungsprozesses identifiziert. Diese Vorarbeiten bilden die Grundlage für die nächste Phase, in der die Entwicklung der Plattform und die Umsetzung der technischen Architektur im Vordergrund stehen.

## 1.2 Stärken

Nr	Beschreibung
01	<b>Zugang zu fortschrittlichen Technologien:</b> Die Entscheidung, die Cloud für die Hosting-Infrastruktur zu nutzen, bietet eine solide Grundlage für Skalierbarkeit und Zuverlässigkeit.
02	<b>Klare Zielgruppe:</b> Die Fokussierung auf Einzelpersonen und kleine Gaming-Communities sorgt für eine zielgerichtete Ansprache und Bedürfniserfüllung.
03	<b>Benutzerzentriertes Design:</b> Die Priorisierung einer intuitiven Benutzeroberfläche unterstützt die Nutzerfreundlichkeit und Zugänglichkeit für Personen ohne umfangreiches technisches Wissen.
04	<b>Sicherheitsbewusstsein:</b> Die Berücksichtigung von Sicherheitsaspekten und Datenschutz von Anfang an ist essenziell für das Vertrauen der Nutzer.
05	<b>Flexibilität in der Serverkonfiguration:</b> Die Möglichkeit für Nutzer, ihre Server nach individuellen Bedürfnissen zu konfigurieren, fördert die Personalisierung und Zufriedenheit.
06	<b>Integration moderner Zahlungsmethoden:</b> Die Implementierung eines einfachen Plugins für Zahlungen (z.B. Twint) vereinfacht den Kaufprozess und erhöht die Benutzerfreundlichkeit.
07	<b>Konzeptionelle Vorarbeiten:</b> Die bereits geleisteten konzeptionellen Überlegungen, insbesondere bezüglich Monetarisierung und technischer Umsetzung, bieten eine solide Ausgangsbasis.
08	<b>Teamzusammensetzung und -motivation:</b> Das engagierte und interdisziplinäre Projektteam bringt eine Vielfalt an Fähigkeiten und Perspektiven in das Projekt ein, was die Innovationskraft stärkt.

## 1.3 Schwächen

Nr	Beschreibung	Beurteilung	Ursache
S1	<b>Mangel an detailliertem technischen Plan:</b> Die Ausgangslage zeigt eine allgemeine Idee und einige Überlegungen zu Technologien, aber es fehlt ein detaillierter technischer Entwurfsplan, der spezifiziert, wie die verschiedenen Komponenten zusammenwirken werden.	Diese Lücke könnte zu Verzögerungen oder Herausforderungen in der Entwicklungsphase führen.	Möglicherweise fehlte es an Zeit oder technischem Know-how in der Anfangsphase.
S2	Begrenzte Sicherheitsplanung: Während Sicherheitsbewusstsein als Stärke erwähnt wurde, gibt es wenig Details darüber, wie Sicherheit konkret umgesetzt wird, insbesondere im Hinblick auf die	Dies stellt ein hohes Risiko für Datenverletzungen und andere Sicherheitsprobleme dar.	Möglicherweise fehlt es an spezifischem Sicherheits-Know-how im Team oder an einer Unterschätzung der

Nr	Beschreibung	Beurteilung	Ursache
	Absicherung der Server und den Schutz von Nutzerdaten.		Sicherheitsanforderungen.
S3	<b>Mangelnde rechtliche Vorbereitung:</b> Es gibt keine Erwähnung der Berücksichtigung rechtlicher Aspekte, insbesondere im Zusammenhang mit Datenschutz, geistigem Eigentum und Nutzervereinbarungen.	Rechtliche Unklarheiten könnten zu Verzögerungen, Strafen oder Einschränkungen in der Funktionalität führen.	Unzureichende rechtliche Expertise im Team oder Unterschätzung der rechtlichen Komplexität von Online-Diensten.

## 2 Ziele

Nr.	Beschreibung	Messgrösse	Priorität	Bezug zu Situation
Z1	Entwicklung eines detaillierten technischen Plans, der die Architektur, die Integration der Komponenten und die Interaktionen zwischen Frontend, Backend und Infrastruktur klar definiert.	Ein umfassendes Architekturdokument ist erstellt, das alle Systemkomponenten und ihre Beziehungen zueinander detailliert beschreibt.	Hoch	Adressiert den identifizierten Mangel an einem detaillierten technischen Plan, der eine klare Vorstellung davon vermittelt, wie die Plattformkomponenten interagieren sollen.
Z2	Entwicklung und Implementierung eines umfassenden Sicherheitskonzepts, das sowohl die Absicherung der Serverinfrastruktur als auch den Schutz von Nutzerdaten umfasst.	Ein Sicherheitskonzept ist erstellt und umgesetzt, inklusive Massnahmen zur Datensicherung, Verschlüsselung und Zugriffskontrolle.	Hoch	Geht auf die in der Situationsanalyse festgestellte Lücke in der Sicherheitsplanung ein, insbesondere im Hinblick auf den Schutz der Serverinfrastruktur und der Nutzerdaten.
Z3	Sicherstellung, dass die Plattform alle relevanten rechtlichen Anforderungen, insbesondere im Bereich Datenschutz und geistiges Eigentum, erfüllt.	Alle gesetzlichen Anforderungen sind identifiziert und in der Plattform umgesetzt, einschliesslich Datenschutzerklärung und Nutzervereinbarungen.	Hoch	Nimmt Bezug auf die in der Ausgangssituation festgestellte mangelnde Berücksichtigung rechtlicher Aspekte, was für die Plattform zu rechtlichen Herausforderungen führen könnte.
Z4	Entwicklung eines intuitiven und ansprechenden Frontend-Designs, das es Nutzern ermöglicht, ohne technische Vorkenntnisse Gameserver zu konfigurieren und zu verwalten.	Durchführung von Usability-Tests mit einer Zielgruppenerfolgsrate von über 90% bei der Durchführung grundlegender Aufgaben wie Accounterstellung, Serverkonfiguration und -verwaltung.	Hoch	Direkt adressiert die Ausgangslage, die eine benutzerfreundliche Oberfläche für technisch nicht versierte Nutzer fordert.
Z5	Integration eines sicheren und benutzerfreundlichen Zahlungssystems, das den Zahlungsprozess für die Nutzer vereinfacht und verschiedene Zahlungsmethoden unterstützt.	Implementierung des Bezahlplugins und evtl. weiterer Zahlungsmethoden mit einer erfolgreichen Transaktionsrate.	Mittel	Baut auf der Erwähnung der Zahlungsabwicklung über das Twint-Plugin auf und erweitert die Funktionalität um zusätzliche Zahlungsoptionen.
Z6	Automatisierung des Prozesses der Serverbereitstellung und -konfiguration, um eine schnelle und fehlerfreie Einrichtung der	Entwickelte und getestete Skripte (z.B. Terraform für die Bereitstellung, CloudInit für die Konfiguration), die	Hoch	Geht auf die vorgeschlagene Verwendung von Terraform und CloudInit ein, um die Bereitstellung und

	Gameserver zu gewährleisten.	es ermöglichen, einen neuen Server innerhalb von Minuten vollständig aufzusetzen und zu konfigurieren.		Konfiguration zu automatisieren.
Z7	Implementierung eines Systems, das es Nutzern ermöglicht, individuelle Subdomains für ihre Gameserver einzurichten und zu verwalten.	Entwicklung einer Funktion, die automatisiert Subdomains im Format id-vom-server.nutzerdomain.com erstellt und konfiguriert, sobald ein Server eingerichtet wird.		Bezieht sich auf die Anforderung, dass Server über spezifische Subdomains zugänglich sein sollen.
Z8	Erfolgreiche Markteinführung der Plattform mit dem Ziel, eine festgelegte Anzahl von aktiven Nutzern und Gameservern innerhalb der ersten sechs Monate zu erreichen.	Erreichen von X aktiven Nutzern und Y gehosteten Gameservern innerhalb von sechs Monaten nach dem Launch.	Mittel	-
Z9	Entwicklung und Implementierung einer Skalierbarkeitsstrategie, um das Wachstum der Nutzerbasis und der Serverlast effizient zu unterstützen.	Die Plattform kann ohne Leistungseinbussen eine Verzehnfachung der Nutzer- und Serverlast bewältigen.	Mittel (Da eher weniger User zum Beginn erwartet sind)	-

\*Orange → Geschäftsziele (unabhängig von Schwachpunkten oder Analyse)

## 2.1 Rahmenbedingungen

**Vorgehensmodell:** Das Projekt wird unter Anwendung des HERMES-gibb Vorgehensmodells durchgeführt, das einen strukturierten Rahmen für Projektmanagement und -abwicklung in der IT bietet.

**Zeitliche Rahmenbedingungen:** Der Zeitaufwand für das Projekt ist auf für die Entwicklung auf etwas 3x4 Lektionen während eines Monats plus zusätzliche Freizeit der Projektmitglieder begrenzt. Dies setzt eine straffe Planung und Priorisierung der Projektaufgaben voraus.

**Räumliche Rahmenbedingungen:** Das Projektteam hat Zugang zu einem Raum in der Gibb (Berufsbildungsschule Bern) für Besprechungen und gemeinsame Arbeitssessions. Zudem wird das WLAN der Gibb genutzt, was die Arbeit vor Ort erleichtert. Zuhause wird nur der Laptop, MS Teams und eine Internetverbindung benötigt.

**Personelle Ressourcen:** Das Projektteam besteht aus den Mitgliedern Maurice Däppen (Projektleiter und Entwickler), Patrick Aeschlimann, Lenny Herren, Mika Hannappel (alle Entwickler) und Christian Kissling (Vertreter des Auftraggebers Valve (Steam)). Dabei werden pro Woche 2 Lektionen in der Gibb und nochmals so viel zuhause genutzt. Es wird mit einem Stundensatz von 80.- gerechnet.

**Kommunikationswerkzeuge:** Für die Kommunikation innerhalb des Teams und mit dem Auftraggeber wird Microsoft Teams verwendet. Zur Dokumentation und zum Austausch von Projektmaterialien kommen Microsoft Word und gegebenenfalls weitere Tools der Microsoft Office Suite zum Einsatz.

## 2.2 Abgrenzung

**Entwicklung der Spiele selbst:** Das Projekt umfasst nicht die Entwicklung oder Modifikation der Spiele, die auf den Servern gehostet werden. Es konzentriert sich ausschliesslich auf die Bereitstellung der Infrastruktur und Plattform zur Serververwaltung.

**Umfassende Kundensupport-Dienste:** Während grundlegende Support-Dienste für die Plattformnutzung bereitgestellt werden können, ist ein umfassender Kundensupport, der über die Plattformnutzung hinausgeht (z.B. spezifische Spielefragen), nicht Teil des Projekts.

**Physische Hardware:** Das Projekt befasst sich nicht mit der Bereitstellung oder Wartung physischer Serverhardware, da die Infrastruktur vollständig in der Cloud gehostet wird.

**Langfristige Betriebsführung:** Die langfristige Betriebsführung und -überwachung der Plattform sowie der gehosteten Gameserver, über die initiale Inbetriebnahme und den Launch hinaus, könnten über den Rahmen dieses Projekts hinausgehen und erfordern möglicherweise separate Betriebsvereinbarungen.

**Rechtliche Beratung:** Während das Projekt die Einhaltung der rechtlichen Anforderungen anstrebt, ist eine spezialisierte rechtliche Beratung oder umfassende rechtliche Prüfung ausserhalb des Projektumfangs.

### 3 Liste der Stakeholder

#### Projektteam:

- Mitglieder: Maurice Däppen (Projektleiter und Entwickler), Patrick Aeschlimann, Lenny Herren, Mika Hannappel (alle Entwickler)
- Rolle: Hauptverantwortliche für die Planung, Entwicklung und Umsetzung des Projekts.

#### Auftraggeber:

- Organisation: Valve (Steam)
- Rolle: Finanzielle und strategische Unterstützung des Projekts, potenzieller Nutzen durch die Integration mit der Steam-Plattform.

**Endnutzer (Kunden) (Denn diese sind für den Erfolg verantwortlich und haben auch was zu sagen, jedoch nur indirekt):**

- Gruppe: Einzelpersonen und kleine Gaming-Communities, die private Gameserver für Spiele wie Minecraft oder Ark hosten möchten.
- Rolle: Direkte Nutzer der Plattform, deren Feedback und Zufriedenheit für den Erfolg des Projekts entscheidend sind.

### 4 Anforderungen (Initial Product Backlog)

Initial Product Backlog		
Anforderungen / Tätigkeiten	Aufwands-schätzung (Story Points)	Begründung
Als Entwickler möchte ich ein detailliertes Architekturdokument erstellen, das alle Systemkomponenten und deren Beziehungen klar definiert, um eine solide Basis für die Entwicklung zu schaffen.	8	Die Erstellung eines umfassenden Architekturdokuments erfordert eine tiefe Analyse der Systemanforderungen und -komponenten, was einen erheblichen Aufwand darstellt.
Als Entwickler möchte ich ein umfassendes Sicherheitskonzept entwickeln und implementieren, das Datensicherung, Verschlüsselung und Zugriffskontrollen umfasst, um die Plattform und Nutzerdaten zu schützen.	13	Die Entwicklung und Implementierung eines Sicherheitskonzepts, das verschiedene Sicherheitsaspekte abdeckt, ist komplex und zeitaufwändig.
Als Entwickler möchte ich sicherstellen, dass die Plattform alle relevanten rechtlichen Anforderungen erfüllt, indem Datenschutzerklärungen und Nutzervereinbarungen entwickelt und implementiert werden.	5	Während die Identifizierung gesetzlicher Anforderungen und die Erstellung entsprechender Dokumente wichtig sind,



Initial Product Backlog		
Anforderungen / Tätigkeiten	Aufwands- schätzung (Story Points)	Begründung
		handelt es sich um eine klar umrissene Aufgabe.
Als UI/UX-Designer möchte ich ein intuitives und ansprechendes Frontend-Design erstellen, das es Nutzern ermöglicht, ohne technische Vorkenntnisse Gameserver zu konfigurieren und zu verwalten.	8	Das Design einer benutzerfreundlichen Oberfläche erfordert Kreativität und mehrere Iterationen, um Benutzerfeedback zu berücksichtigen.
Als Entwickler möchte ich ein sicheres Zahlungssystem integrieren, das den Zahlungsprozess vereinfacht und verschiedene Zahlungsmethoden unterstützt, um den Nutzern Flexibilität zu bieten.	5	Die Integration eines Zahlungssystems, insbesondere wenn vorhandene Plugins verwendet werden, ist eine wohldefinierte Aufgabe, die jedoch sorgfältige Tests erfordert.
Als Entwickler/DevOps-Ingenieur möchte ich den Prozess der Serverbereitstellung und -konfiguration automatisieren, um eine schnelle und fehlerfreie Einrichtung der Gameserver zu ermöglichen.	13	Die Automatisierung der Bereitstellung und Konfiguration von Servern erfordert umfangreiche Skripterstellung und Tests, was einen hohen Aufwand darstellt.
Als Entwickler möchte ich ein System implementieren, das automatisch Subdomains für neue Gameserver erstellt und konfiguriert, um den Nutzern eine einfache Zugriffsmöglichkeit zu bieten.	5	Die Automatisierung des Subdomain-Managements ist technisch weniger komplex und kann oft mit vorhandenen Tools umgesetzt werden.
Als Systemarchitekt möchte ich eine Skalierbarkeitsstrategie entwickeln und implementieren, um das Wachstum der Nutzerbasis und der Serverlast effizient zu unterstützen.	8	Die Entwicklung einer Skalierbarkeitsstrategie erfordert eine umfassende Planung und Tests, um sicherzustellen, dass die Plattform unter Last performant bleibt.

## Definition of Done

### Sicherheit:

- Alle neuen Features müssen Sicherheitsrichtlinien und -best practices folgen, einschließlich Datenverschlüsselung, sicherer Authentifizierung und Autorisierung.
- Sicherheitsscans und -tests müssen durchgeführt und bestanden werden, bevor ein Feature als fertig gilt.

### Skalierbarkeit:

- Die Lösung muss nachweislich in der Lage sein, mit einer vorher festgelegten maximalen Anzahl von Nutzern und gleichzeitigen Anfragen umzugehen, ohne dass es zu Leistungseinbußen kommt.

### Verfügbarkeit und Zuverlässigkeit:

- Die Plattform muss eine definierte Mindestverfügbarkeit aufweisen, z.B. 99,9% Uptime ausserhalb geplanter Wartungsfenster.
- Fehler- und Ausfalltoleranz muss durch entsprechende Tests nachgewiesen werden.

### **Benutzerfreundlichkeit und Zugänglichkeit:**

- Die Benutzeroberfläche muss klare, intuitive und konsistente Navigation und Gestaltung aufweisen.

## **5 Lösungsvarianten**

### **5.1 Variantenübersicht**

#### **Variante 1: Amazon Web Services (AWS)**

Nutzung von AWS als primärem Cloud-Anbieter für Hosting, Speicherung und Rechenleistung. Nutzung von AWS-spezifischen Diensten wie EC2 für virtuelle Server, RDS für Datenbanken und S3 für Speicher.

#### **Variante 2: Google Cloud Platform (GCP)**

Nutzung von Google Cloud Platform für die Bereitstellung der Infrastruktur und Dienste. Einsatz von Google Compute Engine für virtuelle Maschinen, Google Kubernetes Engine für Container-Orchestrierung und Google Cloud Storage für Daten.

#### **Variante 3: Microsoft Azure**

Nutzung von Microsoft Azure als Cloud-Plattform, mit Schwerpunkt auf Azure Virtual Machines für Hosting, Azure SQL Database für Datenbankdienste und Azure Blob Storage für Datenspeicherung.

### **5.2 Beschreibung der Varianten**

#### **Variante 1: Amazon Web Services (AWS)**

##### **Was ist AWS?**

Amazon Web Services ist eine umfangreiche und weit verbreitete Cloud-Plattform, die von Amazon angeboten wird. Sie stellt virtuelle Computer zur Verfügung, auf denen Server laufen können, Speicherplatz im Internet für Dateien und Datenbanken sowie eine Vielzahl anderer Dienste, die man für den Betrieb einer Online-Plattform benötigt.

##### **Wie funktioniert es für xServer?**

Für das xServer-Projekt bedeutet die Nutzung von AWS, dass wir die gesamte notwendige Hardware (wie Server und Speicher) von Amazon mieten können, anstatt sie selbst kaufen und warten zu müssen. Wir können schnell neue Server für unsere Nutzer einrichten, die Leistung nach Bedarf anpassen und sicherstellen, dass die Daten sicher gespeichert sind.

#### **Variante 2: Google Cloud Platform (GCP)**

##### **Was ist GCP?**

Google Cloud Platform ist eine Sammlung von Cloud-Computing-Diensten, die von Google bereitgestellt werden. Ähnlich wie AWS bietet GCP virtuelle Maschinen, Datenbankspeicher und viele spezielle Dienste für Big Data, maschinelles Lernen und mehr.

##### **Wie funktioniert es für xServer?**

Durch die Verwendung von GCP kann das xServer-Projekt von Googles leistungsstarker Infrastruktur und fortschrittlichen Analysetools profitieren. Dies könnte besonders nützlich sein, wenn wir planen, fortschrittliche Analysen oder maschinelles Lernen für Spielestatistiken oder Nutzerverhalten einzusetzen.

#### **Variante 3: Microsoft Azure**

##### **Was ist Azure?**

Microsoft Azure ist eine Cloud-Computing-Plattform von Microsoft, die eine breite Palette von Cloud-Diensten bietet, darunter Lösungen für Computing, Analytik, Speicherung und Networking. Azure ist bekannt für seine Integration in andere Microsoft-Produkte und Dienste.

## Wie funktioniert es für xServer?

Für das xServer-Projekt würde die Wahl von Azure bedeuten, dass wir eng mit vielen anderen Microsoft-Tools und -Diensten, die möglicherweise bereits verwendet werden, integrieren könnten. Dies könnte den Prozess der Einrichtung und Verwaltung unserer Server vereinfachen und die Zusammenarbeit im Team fördern.

## 6 Bewertung der Varianten (Tabelle)

### AWS

Kriterium	Gewicht	Punkte	Total	
Abdeckung der Anforderungen	200%	3	6	AWS erhielt 3 Punkte wegen seines breiten Dienstleistungsangebots, das alle technischen Anforderungen des Projekts abdecken kann.
Realisierbarkeit, Risiken	150%	3	4.5	Auch hier erzielte AWS die Bewertung von 3 Punkten, da die Plattform eine robuste Infrastruktur mit umfangreichen Dokumentationen und Best Practices bietet, was die Projektrisiken minimiert.
Wirtschaftlichkeit	100%	2	2	AWS wurde hier mit 2 bewertet, da es zwar flexible Preismodelle anbietet, aber bei großem Ressourcenverbrauch kostspielig sein kann.
Technischer Support & Community	50%	3	1.5	Mit einer Bewertung von 3 Punkten zeigt sich, dass AWS einen ausgezeichneten Support und eine groaae Community bietet, was besonders wertvoll für die Problembehandlung und den Wissensaustausch ist.
Skalierbarkeit & Flexibilität	150%	3	4.5	AWS erhielt erneut die Höchstbewertung von 3, da es für seine Anpassungsfähigkeit und Skalierbarkeit bekannt ist, was es ermöglicht, Ressourcen dynamisch anzupassen.
Zuverlässigkeit & Verfügbarkeit	200%	3	6	Mit einer Bewertung von 3 Punkten unterstreicht AWS seine hohe Zuverlässigkeit und Verfügbarkeit, unterstützt durch umfangreiche SLAs.
Kostenstruktur & Preismodell	100%	2	2	Die Bewertung von 2 reflektiert, dass die Kostenstruktur von AWS transparent ist, aber bei intensiver Nutzung zu höheren Ausgaben führen kann.
<b>Gesamtbeurteilung</b>			26.5	

Bewertung: 1-3, Gewichtung in %

## GCP

Kriterium	Gewicht	Punkte	Total	
Abdeckung der Anforderungen	200%	2	4	GCP erhielt eine Bewertung von 2, da es eine breite Palette an Diensten bietet, die viele Anforderungen abdecken, jedoch in spezifischen Bereichen hinter AWS zurückbleiben könnte.
Realisierbarkeit, Risiken	150%	2	3	Auch hier wurde GCP mit 2 bewertet, da die Plattform gut dokumentiert und zuverlässig ist, aber ihre geringere Verbreitung im Vergleich zu AWS in bestimmten Regionen oder bei speziellen Diensten zu Risiken führen könnte.
Wirtschaftlichkeit	100%	3	3	GCP erhielt die Höchstbewertung von 3 für seine kosteneffizienten Preismodelle, insbesondere bei langfristiger Nutzung.
Technischer Support & Community	50%	2	1	Mit einer Bewertung von 2 reflektiert dies, dass GCP soliden technischen Support bietet, die Community jedoch möglicherweise nicht so aktiv wie die von AWS ist.
Skalierbarkeit & Flexibilität	150%	3	4.5	GCP erzielte eine Bewertung von 3 für seine Skalierbarkeit und Flexibilität, besonders durch die Integration mit modernen Technologien wie Kubernetes.
Zuverlässigkeit & Verfügbarkeit	200%	2	4	GCP wurde mit 2 bewertet, da es zwar eine hohe Zuverlässigkeit bietet, aber Berichte darauf hindeuten, dass AWS in bestimmten Aspekten der Verfügbarkeit und Zuverlässigkeit überlegen sein könnte.
Kostenstruktur & Preismodell	100%	3	3	GCP erhielt die Höchstbewertung von 3 für sein attraktives Preismodell, insbesondere durch langfristige Rabatte und detaillierte Kostenkontrolle.
<b>Gesamtbeurteilung</b>			22.5	

Bewertung: 1-3, Gewichtung in %

## Azure

Kriterium	Gewicht	Punkte	Total	
Abdeckung der Anforderungen	200%	2	4	Azure erhielt eine Bewertung von 2, da es viele Projektanforderungen mit einer umfangreichen Palette an Diensten abdeckt, jedoch könnten spezialisierte Dienste oder Integrationen weniger umfassend sein als bei AWS.
Realisierbarkeit, Risiken	150%	1.5	2.25	Azure wurde ebenfalls mit 1.5 bewertet, da die Plattform robust und gut dokumentiert ist, aber spezifische Lösungen oder Konfigurationen herausfordernder zu implementieren sein könnten als bei AWS oder GCP.
Wirtschaftlichkeit	100%	2	2	Azure erhielt eine Bewertung von 2, da es wettbewerbsfähige Preise bietet, aber die Kostenoptimierung und das Verständnis der Preisstruktur komplex sein können, besonders für weniger verbreitete Dienste.
Technischer Support & Community	50%	3	1.5	Azure erzielte die Höchstbewertung von 3/3, profitierend von Microsofts umfangreicher Support-Erfahrung und einer großen, aktiven Community, insbesondere in Unternehmensumgebungen.
Skalierbarkeit & Flexibilität	150%	2	3	Mit einer Bewertung von 2 reflektiert dies, dass Azure solide Skalierbarkeits- und Flexibilitätsoptionen bietet, die Integrationsmöglichkeiten mit Nicht-Microsoft-Produkten jedoch eingeschränkter sein könnten.
Zuverlässigkeit & Verfügbarkeit	200%	2	4	Azure wurde mit 2 bewertet, da es zwar eine starke Zuverlässigkeit aufweist, aber vereinzelte signifikante Ausfälle in der Vergangenheit Bedenken hinsichtlich der Verfügbarkeit aufwerfen könnten.
Kostenstruktur & Preismodell	100%	2	2	Azure erhielt erneut eine Bewertung von 2, da das Preismodell flexibel ist, das Verständnis der Gesamtkosten jedoch herausfordernd sein kann, besonders bei komplexeren Diensten.
<b>Gesamtbeurteilung</b>			18.25	

Bewertung: 1-3, Gewichtung in %

## 7 Lösungsbeschreibung

Nach sorgfältiger Bewertung und Vergleich der drei Cloud-Plattformen – Amazon Web Services (AWS), Google Cloud Platform (GCP) und Microsoft Azure – hat AWS mit einer Gesamtbewertung von 26.5 die höchste Punktzahl erreicht, gefolgt von GCP mit 22.5 und Azure mit 18.5. Aufgrund dieser Bewertung und der spezifischen Stärken, die AWS in verschiedenen Kategorien gezeigt hat, wurde entschieden, dass AWS die bevorzugte Lösungsvariante für das xServer-Projekt ist.

### **Begründung für die Auswahl von AWS:**

Die Entscheidung für AWS als Lösungsvariante für das xServer-Projekt basiert auf einer umfassenden Bewertung der Schlüsselkriterien und unterstreicht die Eignung von AWS, eine leistungsstarke, skalierbare und sichere Plattform für das Hosting und die Verwaltung von Gameservern zu bieten.

**Abdeckung der Anforderungen:** AWS bietet die umfangreichste Palette an Diensten und Tools, die alle technischen Anforderungen des xServer-Projekts abdecken, von der Serverbereitstellung über Datenbanken bis hin zu Netzwerksicherheit.

**Skalierbarkeit & Flexibilität:** AWS ermöglicht eine nahtlose Skalierung, was für das xServer-Projekt entscheidend ist, um mit der variierenden Anzahl von Nutzern und Anforderungen umzugehen.

**Zuverlässigkeit & Verfügbarkeit:** AWS' bewährte Zuverlässigkeit und umfangreiche Service Level Agreements (SLAs) bieten eine solide Grundlage für die hohe Verfügbarkeit der Gameserver.

**Technischer Support & Community:** Der umfangreiche technische Support von AWS und eine aktive Entwicklercommunity bieten wertvolle Ressourcen für die Lösung von Herausforderungen und den Wissensaustausch.

**Wirtschaftlichkeit:** Trotz der potenziell höheren Kosten bietet AWS flexible Preismodelle und Dienste wie Reserved Instances und Spot Instances, die helfen können, die Kosten zu optimieren.

### **Beschreibung der AWS-Lösung:**

Die Lösung mit AWS besteht aus mehreren Schlüsselkomponenten, die zusammenarbeiten, um eine robuste, skalierbare und sichere Plattform für das Hosting und die Verwaltung von Gameservern zu bieten:

**Amazon EC2 (Elastic Compute Cloud):** Dient als Rückgrat für die Bereitstellung virtueller Server, auf denen die Gameserver laufen. EC2 ermöglicht eine flexible Skalierung der Rechenressourcen je nach Bedarf der Nutzer.

**Amazon RDS (Relational Database Service):** Bietet verwaltete Datenbankdienste, die für die Speicherung von Spielerdaten, Spielständen und anderen relevanten Informationen genutzt werden können.

**Amazon S3 (Simple Storage Service):** Dient zur Speicherung von Spieldateien, Benutzerinhalten und Backups. S3 gewährleistet hohe Verfügbarkeit und Datenbeständigkeit.

**Amazon VPC (Virtual Private Cloud):** Ermöglicht die Erstellung eines isolierten Netzwerkbereichs innerhalb der AWS-Cloud, um eine sichere Umgebung für die Gameserver zu schaffen.

## **8 Projektplanung**

Siehe separates Dokument Projektplan.