

DESIGN AND EVALUATION OF A SERVER-POWERED FRAMEWORK FOR DATA-DYNAMIC DRIVEN JAPANESE CULTURAL AND LANGUAGE EDUCATION IN FrameVR

A Special Project Presented to the
Faculty of the Department of Computer Science,
University of the Philippines Cebu

In Partial Fulfillment
Of the Requirements for the Degree
Bachelor of Science in Computer Science

Ziedell Alexander C. Figueroa
Bachelor of Science in Computer Science

Roberto B. Figueroa Jr.
Adviser

March 2026



UNIVERSITY OF THE PHILIPPINES CEBU
College of Science
Department of Computer Science

DESIGN AND EVALUATION OF A SERVER-POWERED FRAMEWORK FOR DATA-DYNAMIC DRIVEN JAPANESE CULTURAL AND LANGUAGE EDUCATION IN FrameVR

Permission is given for the following people to have access to this thesis:

Available to the general public	**Yes**
Available only after consultation with author or thesis adviser	**Yes**
Available to those bound by confidentiality agreement	**Yes**

ZIEDELL ALEXANDER C. FIGUEROA
Student

ROBERTO B. FIGUEROA JR.
Special Problem Adviser

APPROVAL PAGE

The Faculty of the University of the Philippines Cebu, Department of Computer Science in the College of Science approves this Special Problem entitled

DESIGN AND EVALUATION OF A SERVER-POWERED FRAMEWORK FOR DATA-DYNAMIC DRIVEN JAPANESE CULTURAL AND LANGUAGE EDUCATION IN FrameVR

By Ziedell Alexander C. Figueroa

Assoc. Prof. Roberto B. Figueroa Jr., PhD SP Adviser	Date signed
Asst. Prof. Dharyll Prince M. Abellana Chair, Department of Computer Science	Date signed
Alvin G. Roxas, M. D. Dean, College of Science	Date signed

ACKNOWLEDGMENT

This thesis would not have been possible without the guidance, support, and encouragement of many individuals and institutions who have shaped both its technical rigor and cultural depth.

First and foremost, I extend my deepest gratitude to my thesis adviser, Dr. Roberto Figueroa Jr., whose expertise in immersive systems and educational technology provided invaluable direction throughout this research. His insights challenged me to refine my architectural approach and remain grounded in both theoretical and practical considerations.

I am sincerely thankful to the Department of Computer Science at the University of the Philippines Cebu, whose courses and mentorship cultivated my passion for modular system design, gamification, and cultural integration. Special thanks to Sir Ryan Dulaca for connecting me to Dr. Roberto Figueroa, and whose discussions on educational technology directly influenced the development of Komprensible.

To the UP Open University and my peers at ISTaR (Immersive Spatial Technologies and Open Realities) Labs, thank you for the internship opportunity at UPOU, where I gained hands-on experience with VR devices and learned how distance learning contributes to a more interconnected society.

To my peers and collaborators at the Japan Foundation Manila, whose belief in my immersive education work allowed me to reimagine how language and culture can be taught through technology. Your perspectives helped me made this system extensible with real-world relevance and cultural sensitivity.

I acknowledge the technical communities and open-source contributors whose tools empowered the backend development of Komprensible (Replit, ExpressJS). The spirit of collaboration in these ecosystems continues to inspire my work.

I am deeply grateful to the Central Visayas Studies Center (CVSC) and the Office of the Chancellor of the University of the Philippines Cebu for the Thesis and Dissertation Grant (TDG) 2026. This financial support enabled me to dedicate the necessary time and resources to complete this research with rigor and quality.

To my family and friends, especially those who patiently listened to my troubles and worries, thank you for your unwavering support. Your encouragement sustained me through long nights of coding, testing, and writing.

Finally, I dedicate this work to the learners and educators who seek deeper cultural understanding through immersive experiences. May Komprensible serve as a stepping stone toward more inclusive, dynamic, and meaningful educational systems.

ABSTRACT

This study explores the development of a modular, API-powered framework for delivering dynamic data-powered Japanese language and cultural education within immersive environments. Leveraging platforms like FrameVR, the system integrates real-time content, such as Kanji of the Day, grammar tips, cultural fact boards, and quizzes, through structured JSON endpoints. The research notes the pedagogical value of gamification and comprehensible input theory, while emphasizing and showcasing how live data can enhance learner engagement, contextual relevance, and scalability in virtual spaces, along with its technical validation, benchmarking, and backend architecture evaluations for extensibility and performance testing. Although the current implementation is limited to text-based assets due to platform constraints, the architecture anticipates future support for multimedia and iframe-based content. Broader applications, such as disaster awareness dashboards, climate data visualizations, industry employee training, distance education settings, among many more, are acknowledged as potential extensions. This project presents the first schema-validated, RESTful backend architecture for dynamic educational content injection into a browser-accessible VR environment, with empirical benchmarking of its performance under simulated concurrent classroom loads. The findings establish a replicable model for API-driven immersive learning systems across language education and beyond.

Keywords*: Gamification, Japanese language learning, cultural education, immersive systems, FrameVR, live API, comprehensible input, modular architecture, virtual reality, JSON automation*

LIST OF TABLES

Table 1. List of API endpoints for the Komprensible Server.....31
Table 2. Evaluation Metrics of the Komprensible Server.....32
Table 3. Evaluation Metrics of the Komprensible Server.....51
Table 4. Latency and Throughput Data.....52

LIST OF FIGURES

Figure 1. Metaversal Approach to Language Learning and Delivery..... 23
Figure 2. Conceptual framework of the VR flow.....25
Figure 3. Overall System Use Case, User Interaction, and FrameVR Integration..... 37
Figure 4. JSON Schemas Used for the Server Data.....38
Figure 5. FrameVR Scene with Word of the Day, Grammar Tip, and Quiz Modules..... 41
Figure 6. FrameVR Scene with the Kanji Factboards..... 42
Figure 7. FrameVR Scene with Cultural Factboards.....42
Figure 8. FrameVR Scene with Japan Foundation and UP Open University Factboards.....43
Figure 9. Specific Flow Diagram for the Quiz API.....44
Figure 10. Login Page (via Admin Key)..... 45
Figure 11. Admin Dashboard (Metrics).....46
Figure 12. Content Management..... 46
Figure 13. API Endpoint Monitoring..... 47
Figure 14. FrameVR Scene Unstarted Quiz..... 54
Figure 15. First Click to the Laptop (10 second response time due to FrameVR)..... 54
Figure 16. Second Click to the Laptop to Reveal Answer (10 second response time)..... 55
Figure 17. Clicking Again Selects Another Random Kanji (10 second response time).....55

TABLE OF CONTENTS

ACKNOWLEDGMENT **4**
ABSTRACT **6**
LIST OF TABLES **7**
LIST OF FIGURES **8**
TABLE OF CONTENTS **9**
INTRODUCTION **11**
1.1 Background of the Study 11
1.2 Statement of the Problem 12
1.3 Objectives of the Study 13
1.3.1 General Objective: 13
1.3.2 Specific Objectives: 13
1.4 Significance of the Study 14
1.5 Scope and Limitations 15
REVIEW OF RELATED LITERATURE AND THEORETICAL FRAMEWORK **17**
2.1 Gamification in Language and Cultural Education 17
2.2 Immersive Systems and VR in Language Learning 18
2.3 Live Data Integration and API-Driven Content Delivery 18
2.4 Cultural Integration and VR Pedagogy 19
2.5 Technical Frameworks and Platform-Specific Design 20
2.6 Educational Gamification Theory and Motivation Models 21
2.7 Theoretical Framework 23
2.8 Conceptual Framework 25
2.8.1 Core Components: 26
METHODOLOGY **28**
3.1 Overview of the Technical Validation Approach 28
3.2 Phase 1: System Design and Development 30
3.2.1 API Architecture 30
3.2.2 Content Rotation Logic 31
3.3 Phase 2: Simulation and Automated Testing 31
3.4 Phase 3: Performance Benchmarking and Evaluation 32
3.4.1 Evaluation Metrics 32
3.4.2 Evaluation Frameworks 33
3.5 Phase 4: Architectural Reflection and Future Scalability 33
3.6 Hypotheses 34
PROTOTYPE DESCRIPTION **35**
4.1 Overview and Design Rationale 35
4.2 System Requirements 35
4.3 Architectural Design 36
4.4 API Design and Schema Contracts 38
4.5 Security Model 39
4.5.1 Public Accessibility and Content Exposure 39

4.5.2 Input Validation and Sanitization..... 39
4.5.3 Platform-Level Protections..... 40
4.5.4 Future Considerations..... 40
4.6 Deployment Environment and Integration..... 40
4.7 System Use Case and Flow..... 43
4.8 Admin Dashboard and Monitoring..... 45
4.9 Limitations..... 48
TECHNICAL EVALUATION AND RESULTS..... 49
5.1 Overview and Evaluation Objectives..... 49
5.2 Evaluation Setup..... 49
5.2.1 Environment..... 49
5.2.2 Workloads..... 50
5.3 Metrics and Targets..... 51
5.4 Results..... 52
5.4.1 Latency and Throughput..... 52
5.4.2 Schema Validation..... 53
5.4.3 Quiz State Accuracy..... 53
5.5 Discussion..... 55
5.6 Summary of Findings..... 58
5.7 Evaluation Constraints and Delimitations..... 58
SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS..... 60
6.1 Summary of the Study..... 60
6.2 Conclusions..... 60
6.3 Ethical Use of AI..... 61
6.4 Recommendations and Future Work..... 62
REFERENCES..... 65

CHAPTER 1

INTRODUCTION

1.1 Background of the Study

The integration of immersive technology and gamified learning has revolutionized how educational content can be experienced. In recent years, platforms like FrameVR have enabled developers and educators to build accessible 3D environments where learners can interact with multilingual content, cultural artifacts, and contextual scenarios without the limitations of physical space (FrameVR, 2025). These platforms are part of a growing trend toward “metaverse-based learning,” which promises spatial engagement, real-time responsiveness, and personalized immersion (Kucher et al., 2022; Marques et al., 2023).

In parallel, the use of live data for real-time content delivery has opened new possibilities for flexible educational systems. Through modular API architecture, developers can inject structured information into immersive spaces such as rotating vocabulary, updating cultural facts, and generating quiz logic dynamically. This transforms virtual learning from static visualization into an adaptive experience (Zhang & Hasim, 2022; Smartico, 2025).

Gamification, meanwhile, has emerged as a robust instructional strategy that enhances motivation and retention by embedding elements such as rewards, progression, and interactive challenges (Deterding et al., 2011; Kapp, 2012). When combined with immersive platforms, gamified approaches can deliver “learning by doing”, allowing users to practice new words, navigate cultural contexts, and develop linguistic confidence in a low-stress, playful environment (Dehghanzadeh et al.,

2021; Wang & Md Arif, 2024). These benefits are especially pronounced in second language acquisition, where the concept of comprehensible input plays a crucial role (Krashen, 1982). Learners need contextualized exposure to understandable language in order to build fluency, and immersive environments provide ideal conditions for delivering this input dynamically.

This project presents Komprensible, a modular, API-fed framework for gamified Japanese language and cultural learning. It integrates endpoints for Kanji of the Day, grammar tips, rotating fact boards, and interactive quizzes, all delivered via FrameVR's JSON-based asset automation. While current content is text-based due to platform limitations, future iterations anticipate support for iframes, audio, and multimedia overlays.

Although broader applications, like crisis dashboards, virtual employee training centers, distance education classrooms, or disaster command and training centers, are viable extensions of the framework, they are only briefly referenced here. The thesis centers firmly on gamified Japanese education as its primary contribution, emphasizing backend automation, cultural storytelling, and pedagogical impact.

1.2 Statement of the Problem

While immersive learning environments continue to grow in popularity, most implementations rely on static data, manual asset placement, and hardcoded interactions. This limits their long-term viability, scalability, and pedagogical adaptability. Despite advances in gamification and educational APIs, there remains a significant gap in the use of live data integration for cultural and language education within immersive platforms like FrameVR.

Furthermore, the field lacks a cohesive framework for delivering structured, rotating, and domain-specific content such as kanji facts, cultural trivia, and grammar overlays, without requiring full-stack scene rebuilding. This creates an opportunity for API-fed systems that can sustain high contextual relevance while reducing development overhead.

Problem Statement: How can a modular API framework be designed to deliver live, gamified, and contextually relevant content into immersive systems for Japanese language and cultural education?

1.3 Objectives of the Study

1.3.1 General Objective:

To develop and evaluate a modular API-powered framework for gamified Japanese language and cultural education in immersive environments, exemplifying the need and future implementations of more live and API-powered VR worlds.

1.3.2 Specific Objectives:

- To implement endpoints for Kanji of the Day, grammar tips, quizzes, and cultural fact boards
- To design gamified learning flows using FrameVR's JSON API automation
- To evaluate the system's modularity, scalability, and technical feasibility
- To explore future adaptability for other domains (e.g., disaster awareness, live VR climate dashboards, more dynamic virtual classrooms for Distance Learning, etc.)

1.4 Significance of the Study

This project is significant in several dimensions: technological, pedagogical, institutional, and cultural.

From a technological standpoint, Komprensible showcases how real-time data can be operationalized within immersive environments. It demonstrates the feasibility of delivering structured content through RESTful endpoints that interact with spatial assets, offering developers a reusable architecture for automated learning experiences. The system highlights JSON as a flexible formatting method and promotes modular endpoint design for scaling across diverse topics (Al-Azawi et al., 2021).

In terms of pedagogical value, this project aligns with cognitive and constructivist theories that emphasize active learning. Gamification fosters motivation through challenge and progression, while immersive environments enable learners to practice language in culturally appropriate contexts (Figueroa Flores, 2015). By fusing comprehensible input theory with VR-based exposure, the system creates conditions conducive to second language acquisition, where learners not only understand meanings but associate them with visual and spatial anchors.

For institutions, especially those involved in public education, cultural diplomacy, or digital outreach, such as the University of the Philippines Open University and The Japan Foundation, the framework offers a strategy for maintaining relevance in an increasingly metaverse-powered ecosystem. It provides a low-cost pathway to building interactive, culturally dynamic content that supports long-term engagement (Tapales, 2023; Japan Foundation, 2025).

On a broader cultural level, the thesis contributes to discussions about how technology can preserve, contextualize, and share heritage. Japan's rich linguistic and visual culture can be rendered into gamified overlays that educate while respecting depth. Fact boards about prefectures, festivals, or kanji origins encourage curiosity while promoting intercultural literacy, which is a priority in both local and global education initiatives (Marques et al., 2023).

Finally, this project lays the groundwork for future applications, including smart tourism, civic education, and environmental awareness, all built atop the same API logic. As immersive platforms evolve to support more media formats and external integrations, Komprensible's architecture can scale accordingly, making it both timely and designed for adaptability.

1.5 Scope and Limitations

This project focuses on the development of a backend framework designed to deliver rotating, gamified Japanese language and cultural content into immersive environments using API-powered automation. It primarily operates on FrameVR, a web-based metaverse platform that supports JSON asset injection and asset scheduling. The content includes text-only endpoints for Kanji of the Day, grammar tips, cultural fact boards, and quizzes which are all rendered within VR scenes as text overlays and in-world dynamically-changing factboards.

The scope of the system is limited to textual interactions due to FrameVR's current lack of support for audio, video, and iframe-based asset generation via API routes. While the framework is modular and anticipates future extensions into multimedia content and dynamic scripting, this implementation remains within the bounds of formatted JSON delivery for educational overlays. Its underlying

foundation is the Babylon.js library, and future updates from their platform may address these current limitations (FrameVR, 2025). Other platforms like A-Frame have also been explored, but albeit with more constraints: the platform only supports HTML text dynamically from an existing website, and not JSON response or custom payloads from a server.

The project is further delimited by its primary domain: Japanese language and cultural learning. It does not aim to assess general educational effectiveness, user behavior, or comparative learning outcomes. Instead, it evaluates the technical feasibility, scalability, and system design of delivering dynamic data through immersive environments. Although, these facets could be further explored in a future project.

Applications in other domains, such as disaster response, climate awareness, and civic dashboards, are mentioned as future adaptations and directions, but are not implemented within the scope of this project. Likewise, integration with other immersive platforms (e.g., A-Frame or Mozilla Hubs) may be explored in future research, but the current build is confined to FrameVR for compatibility, consistency, and proof-of-concept execution only.

While the primary deployment environment is FrameVR's browser-based PC interface, the Komprensible system is also accessible on head-mounted immersive hardware (thanks to FrameVR’s existing features), including the Meta Quest 3 via its built-in WebXR browser. This optional pathway supports future experimental testing in fully embodied VR settings but remains outside the technical scope of this project.

CHAPTER 2

REVIEW OF RELATED LITERATURE AND THEORETICAL FRAMEWORK

2.1 Gamification in Language and Cultural Education

Gamification has become a cornerstone of modern educational design, particularly in language acquisition and cultural literacy. Deterding et al. (2011) define gamification as the use of game design elements in non-game contexts, emphasizing motivation, engagement, and progression. Kapp (2012) expands this by arguing that gamification fosters intrinsic motivation through challenge-based learning environments that simulate real-world tasks.

In second language acquisition (SLA), gamified systems have demonstrated measurable improvements in vocabulary retention, grammar mastery, and learner autonomy (Dehghanzadeh et al., 2021). Tapales (2023) introduces the Reinforcement-Assessment Development Theory, framing gamification as a cyclical process of motivation, feedback, and knowledge construction. Zhang and Hasim (2023) conducted a systematic review showing that gamification significantly improves learner attitudes and performance in ESL/EFL contexts.

Cultural education also benefits from gamified storytelling and interactive simulations. Marques et al. (2023) highlight how gamification enhances empathy, creativity, and cross-cultural understanding in heritage education. Wang and Md Arif (2024) demonstrate how interactive cultural heritage systems improve learner engagement and retention through immersive design. Konstantinov et al. (2018) argue that gamified cultural heritage education turns museums and historical sites into interactive classrooms, fostering deeper emotional connections.

2.2 Immersive Systems and VR in Language Learning

Immersive platforms such as FrameVR, Mondly VR, Talkpal, and Dynamic Languages have redefined how learners engage with language and culture. Kucher et al. (2022) found that high-immersion VR environments significantly improve contextual learning and reduce anxiety in language learners. FluentU (2024) and Dynamic Languages (2025) showcase how VR-based lessons simulate real-life scenarios such as ordering food or navigating public spaces, allowing learners to practice Japanese in culturally authentic contexts.

The Japan Foundation's "Home Visit A1" VR course exemplifies how immersive systems can deliver cultural nuance alongside linguistic instruction. Learners explore Japanese homes, practice etiquette, and engage with spatially embedded vocabulary, all without needing specialized hardware (Japan Foundation, 2025). Talkpal's VR Nihongo platform uses AI-powered feedback and customizable scenarios to teach Japanese language and etiquette in virtual marketplaces, classrooms, and social settings (Talkpal, 2025).

Coto Academy (2022) has also begun hosting metaverse-based Japanese learning events, exploring mixed reality for kanji writing, calligraphy, and language exchange. These initiatives demonstrate how immersive platforms can be tailored to specific cultural domains while maintaining scalability and pedagogical rigor.

2.3 Live Data Integration and API-Driven Content Delivery

Live APIs enable real-time content delivery, transforming static VR environments into dynamic learning ecosystems. FrameVR's JSON-based automation allows developers to inject rotating assets such as Kanji of the Day or

cultural fact boards into immersive scenes (FrameVR, 2025). HogoNext (2024) outlines best practices for building real-time APIs, emphasizing persistent connections, message brokers, and client-side libraries for scalable data streaming.

Google's ARCore Geospatial API and Gemini Live API demonstrate how live data can be used to anchor content to real-world coordinates, enabling location-based learning and cultural overlays (Google AI, 2025; Google Developers, 2025). Cesium's integration with OpenUSD and NVIDIA Omniverse illustrates how geospatial data and immersive systems can be combined for cultural storytelling and simulation (Cutsinger, 2023).

These architectures support modularity, personalization, and adaptive learning, which are key features for gamified education systems. The Komprensible framework builds on these principles, using RESTful endpoints to deliver rotating educational overlays in immersive environments.

2.4 Cultural Integration and VR Pedagogy

Virtual reality has proven effective in fostering cultural integration and reducing acculturation stress. Lin et al. (2024) found that immersive VR environments improve academic performance and cultural adaptation among migrant students. Their project emphasizes the role of VR in expanding cultural boundaries and building global cultural capital.

Marques et al. (2023) note that mobile apps, AR, and VR are increasingly used to safeguard intangible heritage while promoting engagement. Altınpulluk (2021) advocates for culturally relevant gamification designs, suggesting the integration of traditional games and localized narratives to enhance inclusivity and

relatability.

The Japan Foundation's KC Virtual Tour allows learners to explore Japanese institutions and homes through 360-degree VR, fostering cross-cultural empathy and spatial literacy (Japan Foundation, 2025). These examples show how immersive systems can serve as platforms for cultural diplomacy, heritage preservation, and intercultural competence.

2.5 Technical Frameworks and Platform-Specific Design

FrameVR stands out as a browser-based metaverse platform optimized for educational deployment. Its support for JSON asset automation, skybox rendering, and modular scene design makes it ideal for a highly customizable API-driven gamification (FrameVR, 2025). Furthermore, it provides seamless browser support for those who don't have VR headsets yet, ensuring accessibility and cross-platform communication between the world's participants. Cesium's open-source extension for Omniverse enables developers to stream 3D Tiles and real-world digital twins at global scale, enhancing cultural simulations and geospatial storytelling (Cutsinger, 2023).

Talkpal's VR Nihongo platform uses GPT-powered AI to deliver personalized feedback and immersive scenarios, while Mondly VR offers speech recognition and chatbot-driven lessons in realistic settings (Mondly, 2025). These platforms demonstrate how immersive systems can be tailored to specific cultural domains while maintaining scalability and pedagogical rigor. However, a common limitation across existing implementations is their reliance on static, hardcoded content that requires manual updates or full scene rebuilds to refresh. Mondly VR and Talkpal,

while pedagogically rich, deliver content through proprietary backends inaccessible to institutional customization. The Japan Foundation's VR course, though openly accessible, serves fixed 360-degree content without dynamic rotation or real-time state management. Komprensible addresses this gap not by competing on content production quality but by offering an open, API-driven delivery architecture that prioritizes content scalability, daily rotation, and institutional adaptability, which are capabilities absent from existing browser-accessible immersive language learning platforms.

2.6 Educational Gamification Theory and Motivation Models

Gamification is grounded in psychological theories of motivation, including Self-Determination Theory (Ryan & Deci, 2000), ARCS (Attention, Relevance, Confidence, Satisfaction) model (Keller, 1987), and Gardner’s distinction between instrumental and integrative motivation (Gardner & Lambert, 1972). Tapales (2023) and Smartico (2025) emphasize how gamification taps into intrinsic motivation by providing clear goals, feedback loops, and emotional engagement.

Celasun and Kaya (2024) argue that gamification fosters cognitive engagement, reduces anxiety, and enhances collaboration. Their project highlights tools like Kahoot, Quizizz, and Duolingo as examples of gamified platforms that make challenging subjects more accessible and enjoyable. These insights inform the design of Komprensible’s learning flows, which use rotating content, quizzes, and cultural overlays to sustain learner interest and retention.

2.7 Theoretical Framework

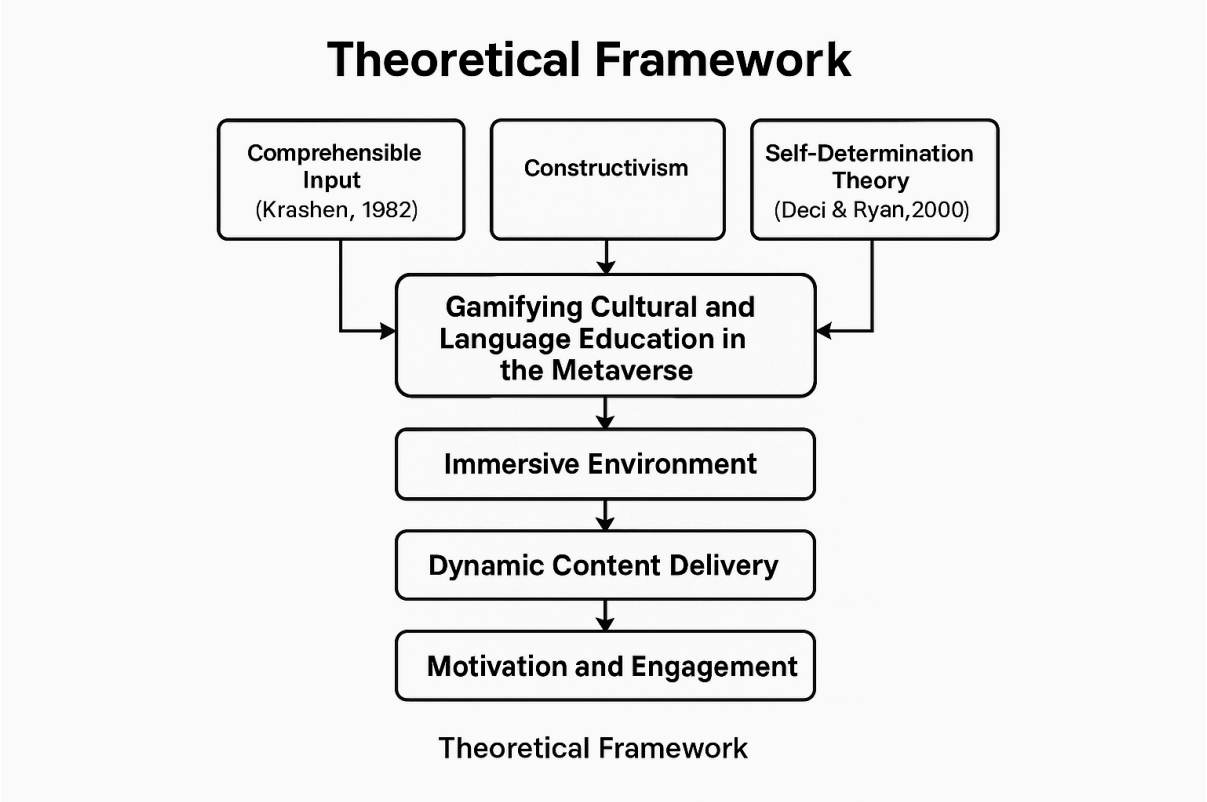


Figure 1. Metaversal Approach to Language Learning and Delivery

This project integrates multiple learning theories to support a live-data-powered, gamified Japanese language and cultural education system deployed within immersive platforms like FrameVR. Unlike behavioral models such as the Technology Acceptance Model (Davis, 1989), which focus on user adoption, this framework emphasizes instructional architecture, data modularity, and the pedagogical affordances of spatial interaction.

At the heart of the theoretical model lies Stephen Krashen's (1982) Comprehensible Input Hypothesis, which posits that language acquisition occurs best when learners are exposed to slightly more advanced but understandable linguistic input ("i+1"). In immersive environments, this input is scaffolded not by a teacher, but via responsive digital overlays triggered through API endpoints, offering

daily kanji, grammar tips, and cultural facts organized by JLPT level as a design-level approximation of progressive scaffolding. Full implementation of adaptive i+1 delivery, where content difficulty responds to individual learner performance, is acknowledged as a direction for future work.

Complementing this, the framework draws on Constructivism, which holds that learners construct knowledge through exploration and contextual experiences (Vygotsky, 1978; Jonassen, 1991). In the Komprensible system, learners navigate Japanese-themed virtual spaces such as homes, shrines, or stations, where content is embedded contextually and activated via API-fed mechanics like quizzes or rotating banners.

Cognitive Load Theory (Sweller, 1988) further guides the future of the interface design, advocating minimal cognitive interference. Gamified elements (e.g., badges, quests, narrative mascots) are intentionally lightweight and modular to avoid overwhelming the learner, while still maintaining engagement through visual cues and micro-rewards (Celasun & Kaya, 2024). After the initial server deployment, this feature is intended to be integrated to theoretically minimize cognitive load.

Finally, Embodied Cognition (Wilson, 2002) and Multimodal Immersion theories support the spatial and interactive nature of the platform. Although the current implementation is text-only, the model anticipates future support for auditory feedback and gesture-triggered overlays, grounding the learning experience in sensory interaction even within virtual worlds.

In sum, this framework operationalizes gamified pedagogical delivery through four pillars: input scaffolding (Krashen), spatial interaction (Constructivism), interface efficiency (Sweller), and sensorimotor anchoring (Wilson), each harmonized within a

modular backend ecosystem.

2.8 Conceptual Framework

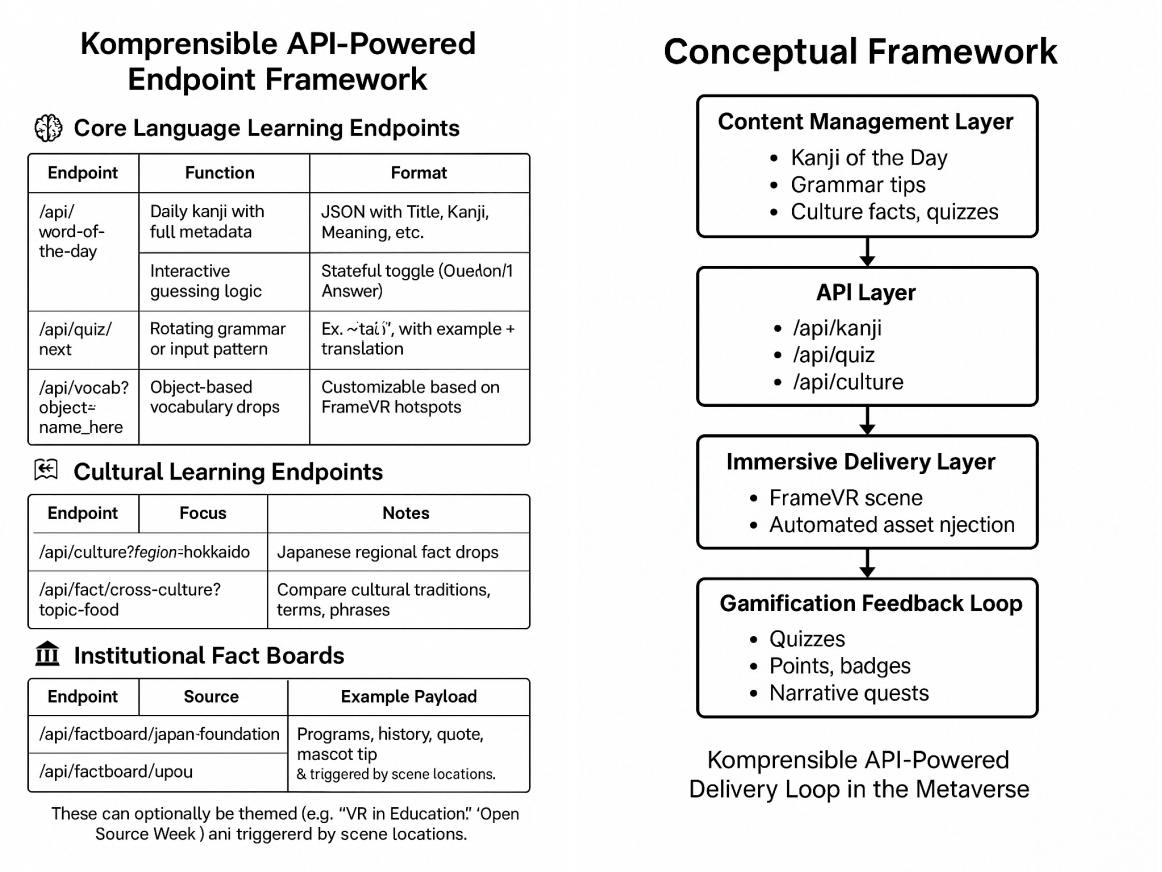


Figure 2. Conceptual framework of the VR flow

This conceptual framework illustrates the operational backbone of Komprensible, a modular system designed to deliver rotating, educational content to immersive environments using live API endpoints. The model does not only simulate passive language and culture exposure; it also orchestrates real-time, gamified linguistic and cultural input within a virtual Japanese ecosystem.

At the core is a Content Scheduler & API Service, which manages text databases for daily kanji, grammar rules, quizzes, and cultural facts. These datasets are formatted into JSON payloads and delivered through structured RESTful routes.

Each endpoint (e.g., `/api/kanji`, `/api/quiz`) cycles content dynamically, allowing for day-based rotations, seasonal themes, or level-specific scaffolding.

The FrameVR Integration Layer parses these API responses and injects them as visual and spatial assets such as banners, overlays, factboards, and mascot dialogues, directly into the virtual environment. Users explore themed zones (e.g., shrines, cross-cultural boards, festival areas), and these zones trigger contextual content delivery.

Though Komprensible's current implementation is text-only, the framework is extensible, anticipating future support for audio playback, iframe-based panels, and intelligent quizzes with branching dialogue.

The conceptual model demonstrates how:

- Comprehensible Input is scaffolded dynamically via APIs
- Gamification guides learner flow and motivation
- Modular endpoints separate content logic from VR design
- Cultural zones link language to identity and social context

This architecture provides not just a learning system but a living content loop that evolves with time, learner needs, and platform capabilities, up to the capabilities of the user's exploration.

2.8.1 Core Components:

1. Content Management Layer
    - Text databases for Kanji of the Day, grammar tips, culture facts, and quizzes

    - JSON payload formatting for endpoint response logic
2. API Layer
    - RESTful routes (`/api/kanji`, `/api/quiz`, `/api/culture`, `/api/grammar-tip`, `/api/vocab`, `/api/fact`, `/api/factboard`) serve content to FrameVR using automated PATCH or POST asset logic
    - Daily rotation mechanisms handle temporal scaffolding of content
3. **Immersive Delivery Layer**
    - FrameVR parses API responses and injects data into spatial scenes via banners, zones, or avatar interactions
    - Visual cues signal learners to interact with content (e.g., quizzes tied to shrine visits or mascots prompting grammar tips, or factboards containing institutional and live updated daily data)
4. **Gamified Feedback Loop**
    - Points, badges, and narrative quests guide exploration (future implementation)
    - Progress feedback may be stored locally or tracked through optional analytics microservices (future implementation)

This conceptual model allows for content scalability (adding new languages, regions, facts), platform modularity (swapping FrameVR for A-Frame or WebXR), and future adaptation for other domains (disaster dashboards, employee trainings, civic tools, training programs) using the same backend logic.

CHAPTER 3

METHODOLOGY

3.1 Overview of the Technical Validation Approach

This project adopts a Design-Based Technical Evaluation (DBTE) methodology to develop and assess the Komprensible framework: a modular, API-powered system for gamified Japanese language and cultural education in immersive environments. Unlike user-centered methodologies that require ethical clearance for human participation, DBTE focuses on system architecture, endpoint behavior, and performance benchmarking through simulation, automated testing, and theoretical alignment. The theories previously mentioned informed endpoint and content design, but were not themselves subject to empirical measurement.

The methodology follows four phases:

1. System Design and Development
2. Simulation and Automated Testing
3. Performance Benchmarking and Evaluation
4. Architectural Reflection and Future Scalability

This approach ensures that the system is not only pedagogically aligned but also technically robust, scalable, and ready for future deployment.

For reference, the test FrameVR world is set upon https://framevr.io/jfm, which is the world developed by the author when they were an intern for Japan Foundation Manila's Metaverse Initiative for Japanese Culture and Language Education, as implied by an agreement between them and UP Open University, also spearheading VR initiatives and Distance Learning. The backend server for delivering dynamic

content can be accessed as view-only in this Replit repository for fast deployment and testing:

https://replit.com/@Ziedell-Alexand/Komprensible-20?replId=d440c5e2-911d-4155-996f-0b837c72f78a

3.2 Phase 1: System Design and Development

3.2.1 API Architecture

Komprensible is built on a RESTful microservice architecture, with endpoints structured to deliver rotating educational content. Each endpoint is designed for modularity, allowing independent updates and content injection into FrameVR scenes.

Core Endpoints:

Category	**Endpoint**	**Function**
Language	`/api/word-of-the-day`	Kanji metadata
Language	`/api/grammar-tip`	Grammar scaffolds
Language	`/api/quiz/next`	Stateful quiz logic
Language	`/api/vocab?object=x`	Object-based vocabulary
Culture	`/api/culture?region=x`	Regional fact drops
Culture	`/api/fact/cross-culture?topic=x`	Filipino-Japanese comparisons
Institutional	`/api/factboard/upou`	UPOU insights
Institutional	`/api/factboard/japan-foundation`	Japan Foundation facts

Optional	`/api/listen?id=x`	TTS (pending platform support)

Table 1. List of API endpoints for the Komprensible Server

Each endpoint returns structured JSON payloads optimized for FrameVR's asset automation engine.

As the Komprensible system is web-native and lightweight, and can be accessible via a PC web browser with traditional game-like controls, it remains compatible with standalone immersive hardware such as the Meta Quest 3. Although formal testing using this device is not part of this project, its accessibility via WebXR browsers enables future deployment in headset-based educational trials.

3.2.2 Content Rotation Logic

Content is rotated daily using a scheduler based on:

- Date hashing
- Index cycling
- Optional seasonal overrides (e.g., Tanabata, Obon)

This ensures freshness and contextual relevance without manual updates.

3.3 Phase 2: Simulation and Automated Testing

To validate endpoint behavior and system integrity, the following tools and strategies will be used:

- Postman: For endpoint simulation, schema validation, and response benchmarking

- JMeter: For load testing and throughput analysis
- JSON Schema Validator: To ensure payload consistency
- Replit Logs / Local Logging: For tracking rotation accuracy and error handling

Test Scenarios:

- Daily rotation accuracy across all endpoints
- Asset injection success rate in FrameVR
- Quiz toggle logic and state persistence
- Fault tolerance under simulated traffic spikes

3.4 Phase 3: Performance Benchmarking and Evaluation

3.4.1 Evaluation Metrics

Metric	**Description**	**Target**
Response Time	Time to first byte (TTFB)	< 200 ms (p95), < 400 ms (p99)
Payload Integrity	JSON schema validity	100% compliance
Uptime	Endpoint availability	> 99.9% over 7 days
Rotation Reliability	Daily content switching	100% accuracy
Quiz State Accuracy	Toggle and passive sync	100% fidelity
Error Rate	Failed requests (5xx)	< 0.1%

Table 2. Evaluation Metrics of the Komprensible Server

These metrics will be monitored using automated tools and logged for future

integration into the Komprensible dashboard.

3.4.2 Evaluation Frameworks

To guide the evaluation, the project draws from:

- GATUGU Framework: Six perspectives for gamified system evaluation (Swacha et al., 2023)
- appRaiseVR: A high-level immersive experience evaluation model (Wienrich & Gramlich, 2020)
- API Performance Benchmarks: Response time, throughput, and fault tolerance standards (Crudu & MoldStud, 2025)

These frameworks ensure that Komprensible is assessed not only for technical performance but also for educational relevance and immersive delivery.

3.5 Phase 4: Architectural Reflection and Future Scalability

Following evaluation, the system will be reviewed for:

- Scalability: Ability to support new domains (e.g., disaster awareness, civic dashboards)
- Extensibility: Integration of multimedia (audio, iframe panels)
- Modularity: Endpoint independence and reusability
- Platform Portability: Potential migration to Unity, A-Frame, or Mozilla Hubs

This reflection phase aligns with Reflective Practice Theory (Schön, 1983), which emphasizes iterative learning through structured inquiry and system feedback. Within this paradigm, developers are encouraged to engage in reflection-on-action by examining design decisions, system performance, and architectural outcomes

post-deployment and *reflection-in-action*, where real-time system behavior informs adaptive refinements. For Komprensible, this means regularly assessing endpoint behavior, rotation logic, and integration fidelity not just in terms of technical correctness, but in how well they align with pedagogical goals like comprehensible input and gamified immersion. By institutionalizing reflection as a design principle, Komprensible positions itself for continual evolution, absorbing feedback from simulated testing, system logs, and future integrations into civic, industry, training, distance education, or crisis-based extensions.

3.6 Hypotheses

- H1: Komprensible's endpoints will deliver content with <200 ms response time under normal load.
- H2: JSON payloads will maintain 100% schema integrity across all endpoints.
- H3: Daily rotation logic will function with 100% reliability over a 7-day test cycle.
- H4: The system will demonstrate modular scalability for future educational domains.

.

CHAPTER 4

PROTOTYPE DESCRIPTION

4.1 Overview and Design Rationale

This chapter presents the architectural design and implementation of Komprensible, a modular backend system developed to deliver dynamic Japanese language and cultural content to immersive environments such as FrameVR. The system was conceived in response to the limitations of existing immersive learning platforms, which often rely on static content, hardcoded interactions, and tightly coupled interface logic. These constraints hinder scalability, reusability, and cross-platform integration (Fielding & Taylor, 2002; Radianti et al., 2020).

Komprensible adopts a backend-first approach, decoupling content orchestration from the immersive interface. It leverages RESTful principles, JSON Schema validation, and observability tooling to ensure content integrity, performance, and maintainability across deployments. The system is designed to be lightweight, extensible, and compatible with both browser-based and head-mounted immersive platforms, including Meta Quest 3 via WebXR.

4.2 System Requirements

The system was designed to fulfill both functional and non-functional requirements. Functionally, it serves structured lesson modules, quizzes, cultural factboards, and media metadata through versioned endpoints. These endpoints support multilingual content, particularly Japanese and English, and enable deterministic content assembly for immersive scenes rendered in FrameVR.

Non-functional requirements include strict performance targets, such as

maintaining latency thresholds of ≤200 milliseconds at the 95th percentile and ≤400 milliseconds at the 99th percentile. Reliability is also critical, with an uptime goal of at least 99.9% and an error rate below 0.1%. Security is enforced through input validation, DDoS rate limiting. Portability is ensured through adherence to web standards and compatibility with WebXR. Maintainability is achieved through a modular codebase, schema governance, and continuous integration pipelines.

4.3 Architectural Design

Komprensible follows a modular architecture inspired by microservice principles (Newman, 2015). The system comprises several core components: an API gateway that routes requests and applies rate limits; a content service responsible for retrieving lesson, quiz, and factboard data; a schema registry that stores and validates JSON Schemas for each content type; a cache layer that implements in-memory caching with ETag support; and a persistence layer that uses a document-oriented database for flexible content storage.

Observability is built into the system through structured logging, metrics collection, and distributed tracing. A CI/CD pipeline automates deployment, schema validation, and contract testing. This architecture ensures that content delivery is both reliable and scalable, while maintaining strict adherence to schema contracts and performance benchmarks.

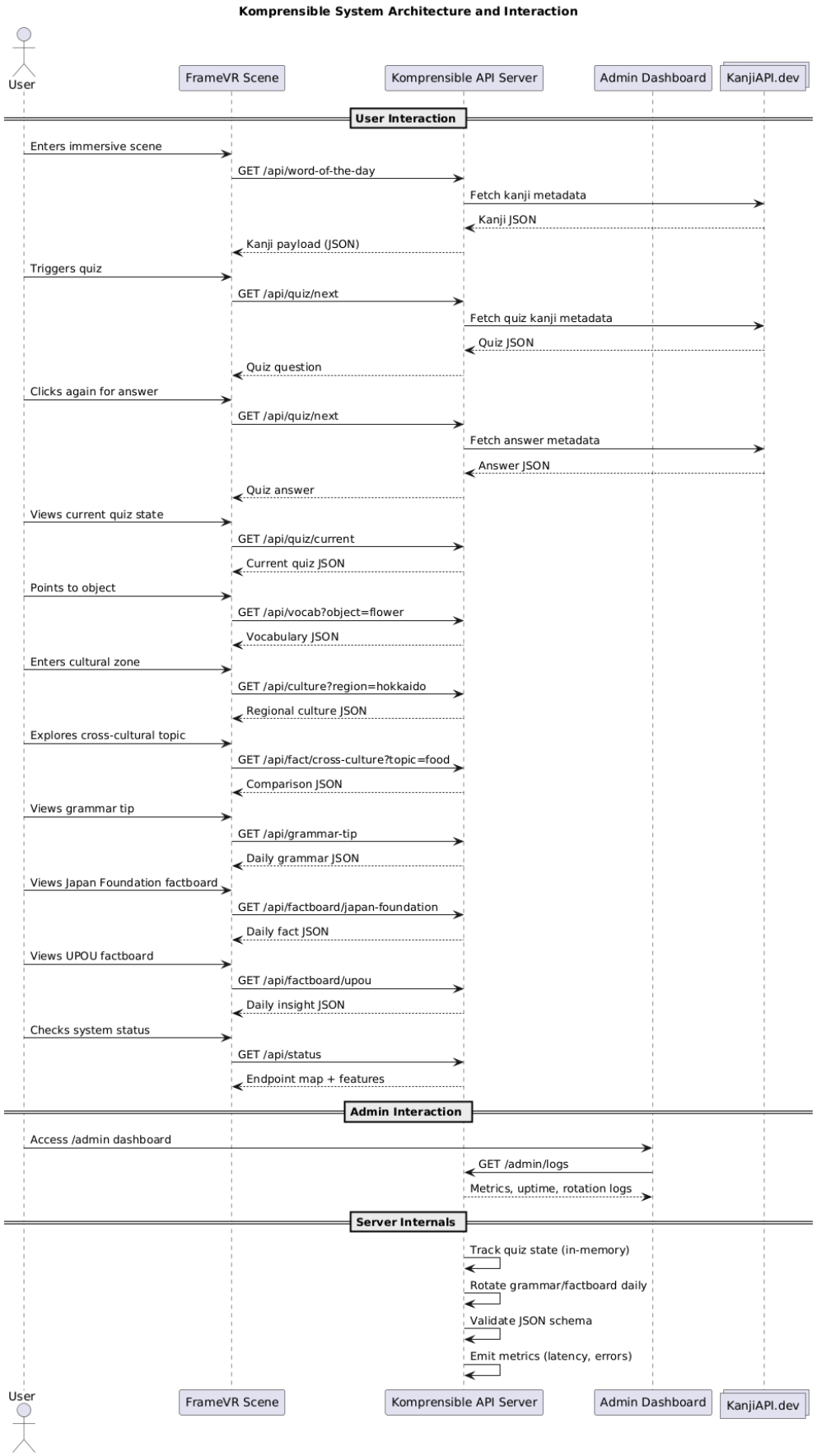


Figure 3. Overall System Use Case, User Interaction, and FrameVR Integration

4.4 API Design and Schema Contracts

The Komprensible API adheres to RESTful conventions and is documented using the OpenAPI 3.1 specification (OpenAPI Initiative, 2021). Each endpoint is passed through an “api” middleware (e.g., `/api/word-of-the-day`) and returns content validated against a corresponding JSON Schema (Bray, 2017). This schema-first approach ensures that all payloads conform to a predefined structure, reducing the risk of integration errors and enabling automated contract testing.

Error handling follows the `application/problem+json` format as defined in RFC 7807 (2016), providing standardized error codes and remediation hints. Responses include `ETag` and `Cache-Control` headers to support efficient caching and content versioning. Content updates trigger cache invalidation via semantic versioning, ensuring that clients always receive the most current data.

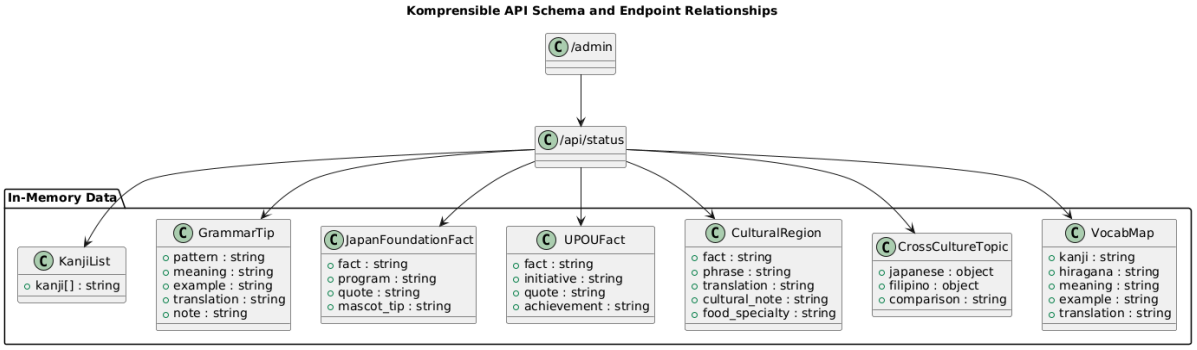


Figure 4. JSON Schemas Used for the Server Data

4.5 Security Model

Komprensible is deployed on Replit as a public-facing backend system, serving educational content without requiring user authentication. This design choice aligns with its intended use in immersive environments such as FrameVR, where content is consumed anonymously and dynamically. While the system does not implement traditional authentication or authorization mechanisms, several safeguards are in place to ensure operational integrity and protect against misuse.

4.5.1 Public Accessibility and Content Exposure

All endpoints are publicly accessible and serve non-sensitive educational data, including Japanese language modules, cultural factboards, and quiz content. Given the absence of user-specific data or private resources, authentication is not required. This simplifies integration with immersive platforms and reduces system complexity.

While the system is not open-source in the strict sense, i.e., the source code is not published under a formal license, it is open-access, allowing any client to retrieve content without credentials. Should future iterations include user-generated content or administrative controls, authentication layers may be introduced.

4.5.2 Input Validation and Sanitization

To prevent injection attacks and malformed requests, all incoming payloads are validated against strict JSON Schemas. This schema-first approach ensures that only well-formed data is processed, reducing the risk of runtime errors and content corruption. Additionally, input sanitization is applied to query parameters and headers to mitigate common web vulnerabilities such as cross-site scripting (XSS)

and header injection.

4.5.3 Platform-Level Protections

Replit provides baseline infrastructure protections including DDoS mitigation, sandboxed execution, and request throttling (Replit, 2025). At the application level, the system mitigates several risks identified in the OWASP API Security Top 10, including injection through strict JSON Schema validation, and excessive data exposure through read-only public endpoints serving only non-sensitive educational content. Risks such as authentication bypass and rate limiting at the application layer are acknowledged as out of scope for the current prototype and are recommended for future iterations targeting institutional deployment. While the code is open-access and can be forked by anyone, the main server operations can only be done by authenticated admins.

These platform-level safeguards help ensure availability and resilience against volumetric attacks. While rate limiting is not explicitly configured at the application level, Replit's default protections offer a reasonable degree of defense for public-facing educational APIs.

4.5.4 Future Considerations

As Komprensible evolves to support administrative dashboards, content authoring tools, or user-specific features, additional security layers such as token-based authentication, role-based access control, and HTTPS enforcement may be introduced. These enhancements would align the system with OWASP ASVS Level 1 (OWASP, 2021) standards for public APIs.

4.6 Deployment Environment and Integration

Komprensible is deployed on a containerized runtime using infrastructure-as-code provisioning. Three environments: development, staging, and production, are maintained to ensure parity and safe rollouts. Deployment follows a blue-green strategy, allowing for seamless updates and rollback capabilities.

Integration with FrameVR is achieved through stateless GET requests that retrieve content for scene assembly. CORS headers are configured to allow access from FrameVR's domain, and content-addressable identifiers are used to ensure deterministic scene builds. This integration pattern allows immersive environments to consume dynamic content without requiring full-stack scene rebuilding.

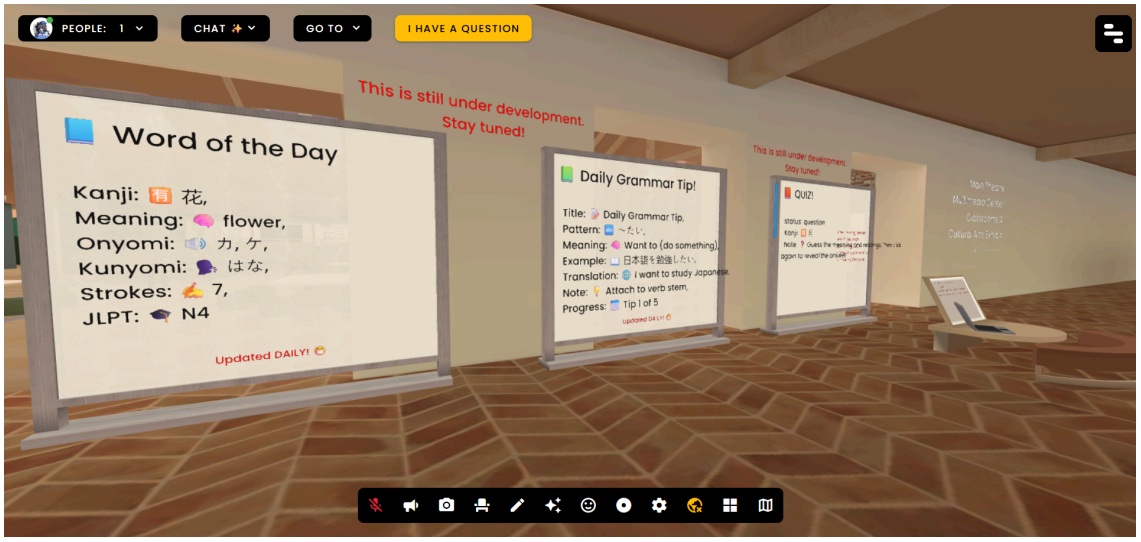


Figure 5. FrameVR Scene with Word of the Day, Grammar Tip, and Quiz Modules

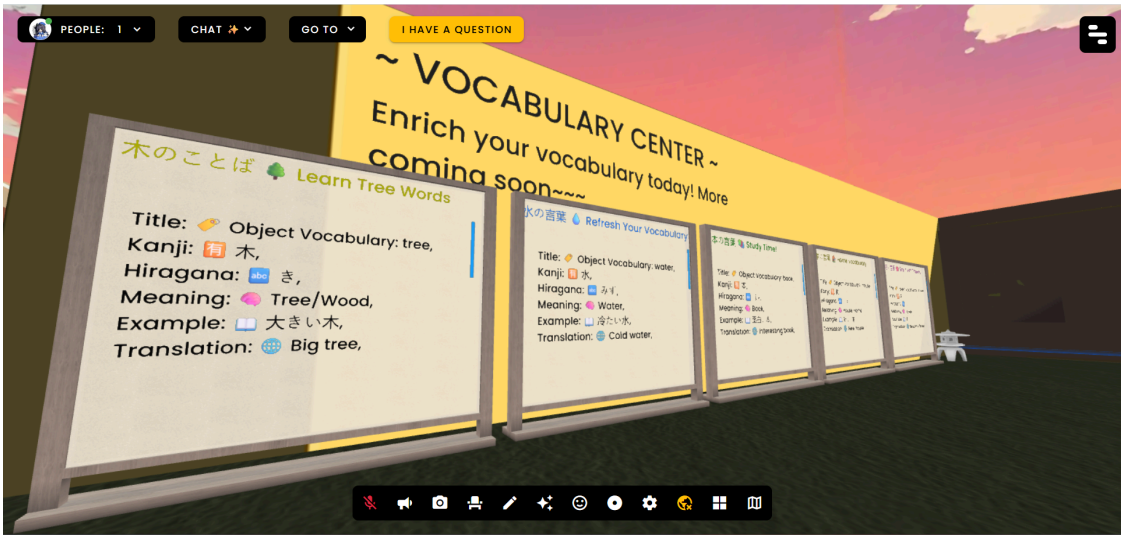


Figure 6. FrameVR Scene with the Kanji Factboards

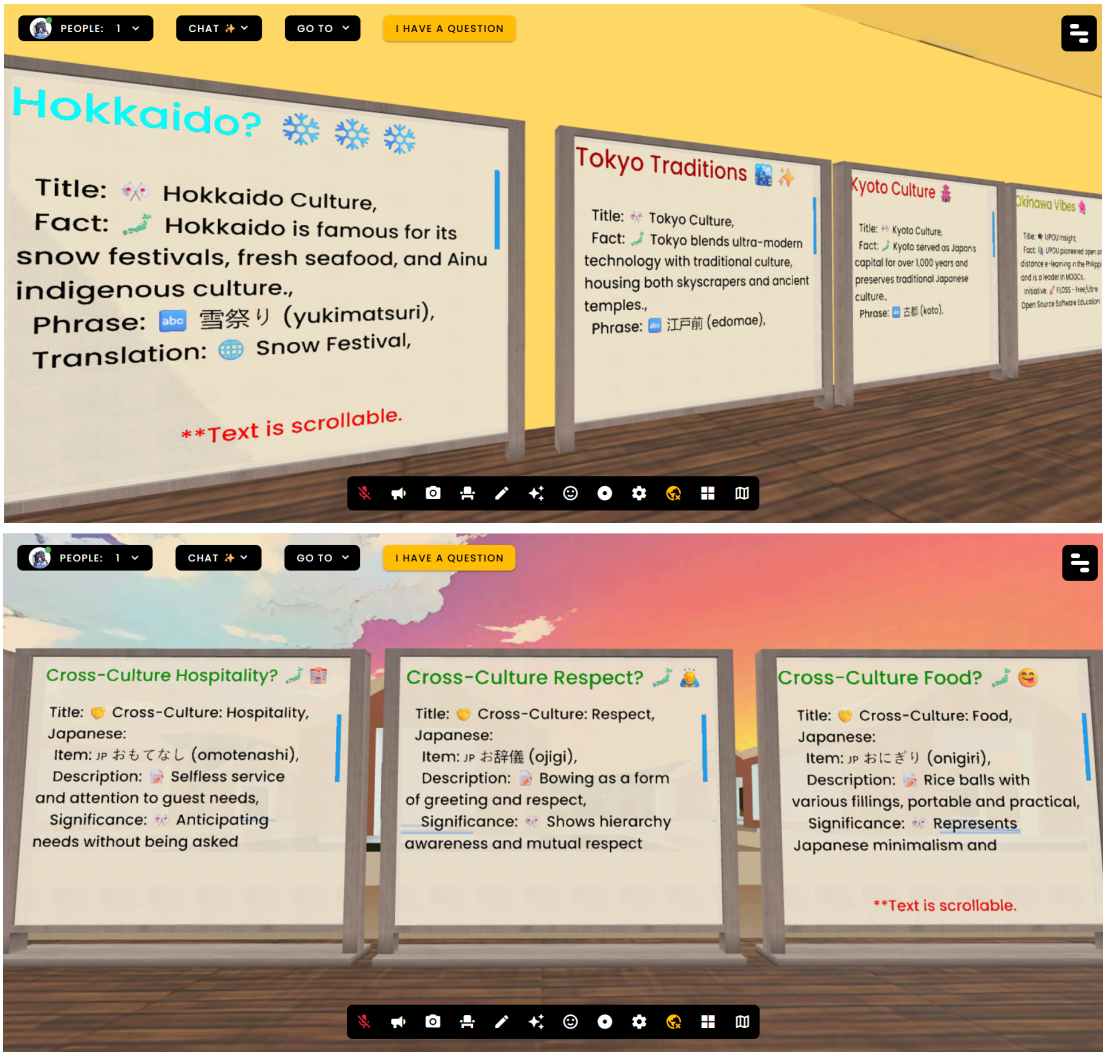


Figure 7. FrameVR Scene with Cultural Factboards

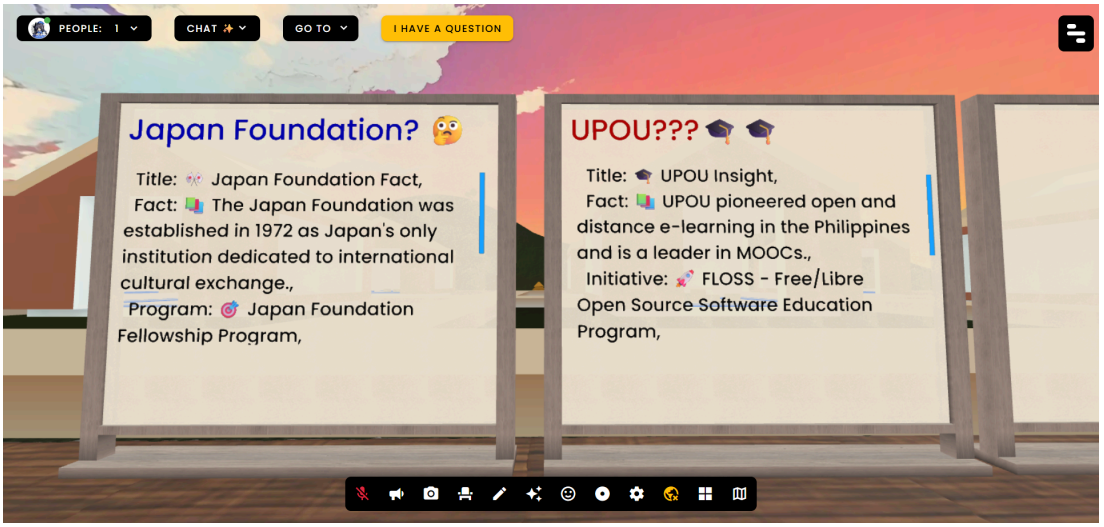


Figure 8. FrameVR Scene with Japan Foundation and UP Open University Factboards

4.7 System Use Case and Flow

The system supports multiple use cases, including daily kanji rotation, grammar tip delivery, cultural factboard injection, and quiz toggling. Each use case begins with a client request to a specific endpoint, followed by schema validation, content retrieval, and response formatting. The response is then consumed by FrameVR's asset automation engine and rendered as a spatial overlay. The overall interaction and system design diagrams can be found in Figure 3. Below in Figure 8 is a more specific snippet of the api/quiz/next user flow that uses state toggling to simulate a simple Kanji quiz.

In the future, more functionality can be added using this state toggling technique via API endpoints, such as a multiple choice quiz that toggles a question and a correct choice, with each choice represented by an item in FrameVR that then toggles that choice via the API to the server handling the quiz, with a passive endpoint that merely shows the question, and shows if an answer was guessed right.

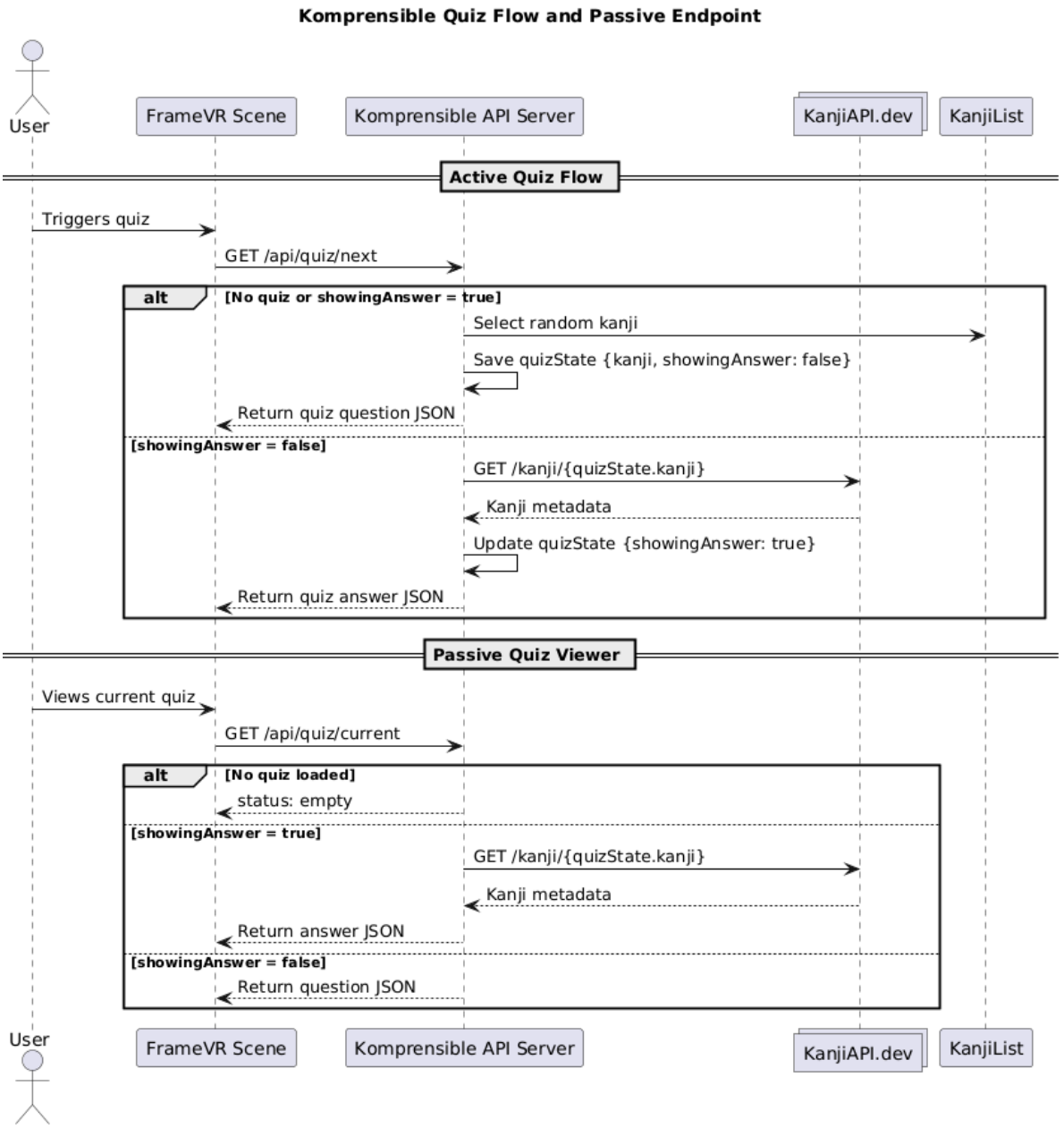


Figure 9. Specific Flow Diagram for the Quiz API

4.8 Admin Dashboard and Monitoring

To support system observability and performance benchmarking, Komprensible includes a lightweight admin dashboard with a simple admin key login access (via .env secrets). This dashboard displays real-time metrics such as endpoint latency, uptime, error rates, and schema validation pass rates. It also logs rotation history and cache performance, enabling developers to monitor system health and content delivery accuracy.

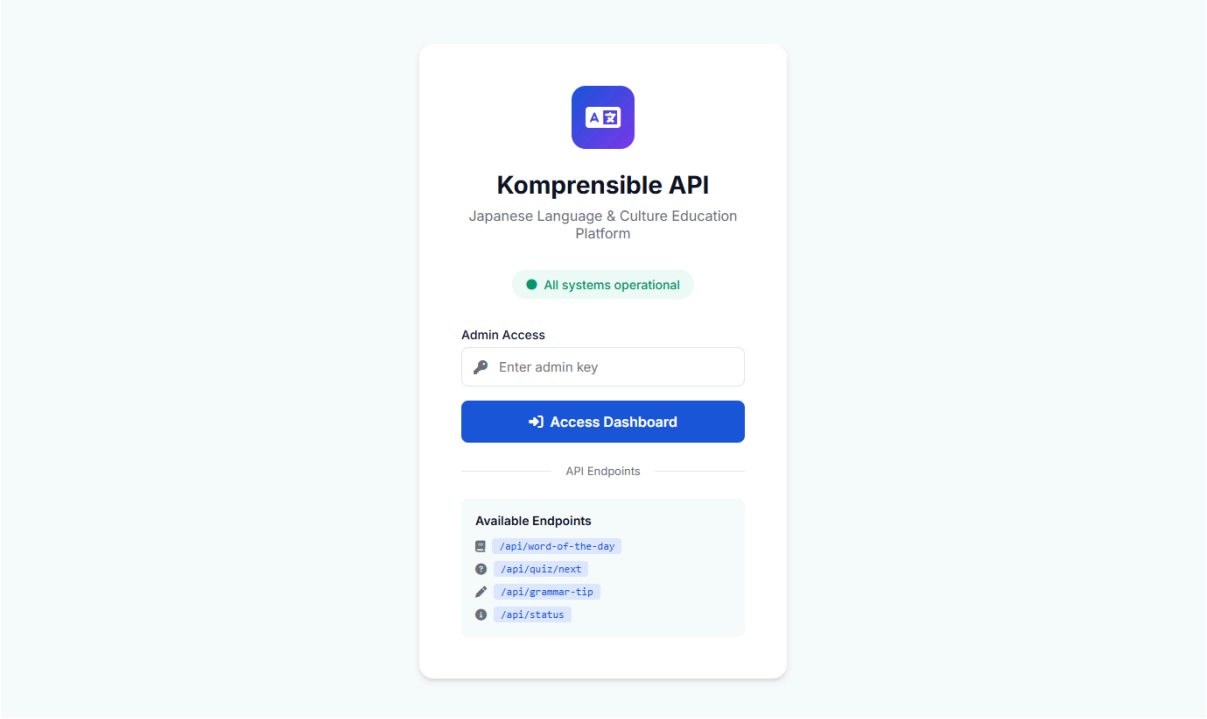


Figure 10. Login Page (via Admin Key)

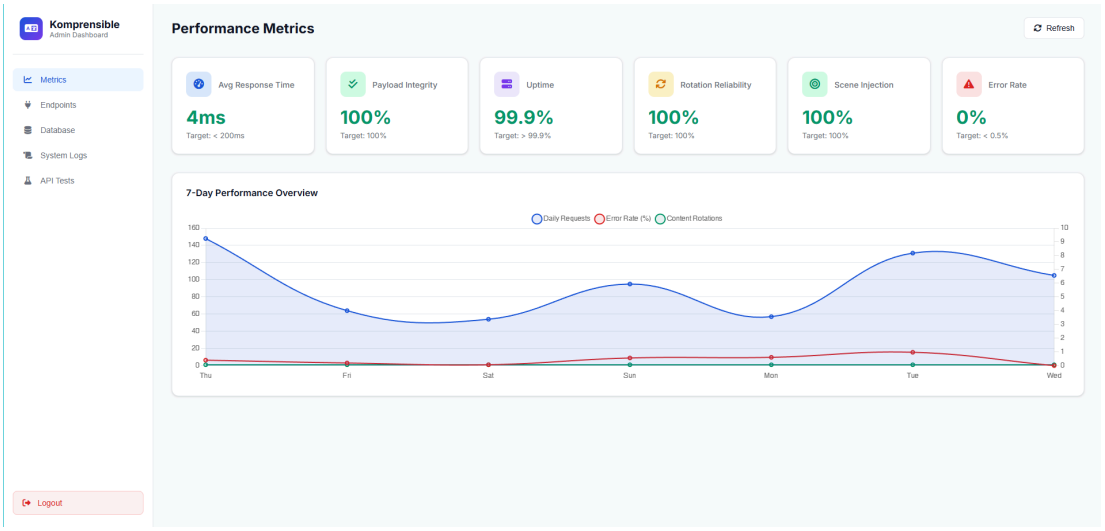


Figure 11. Admin Dashboard (Metrics)

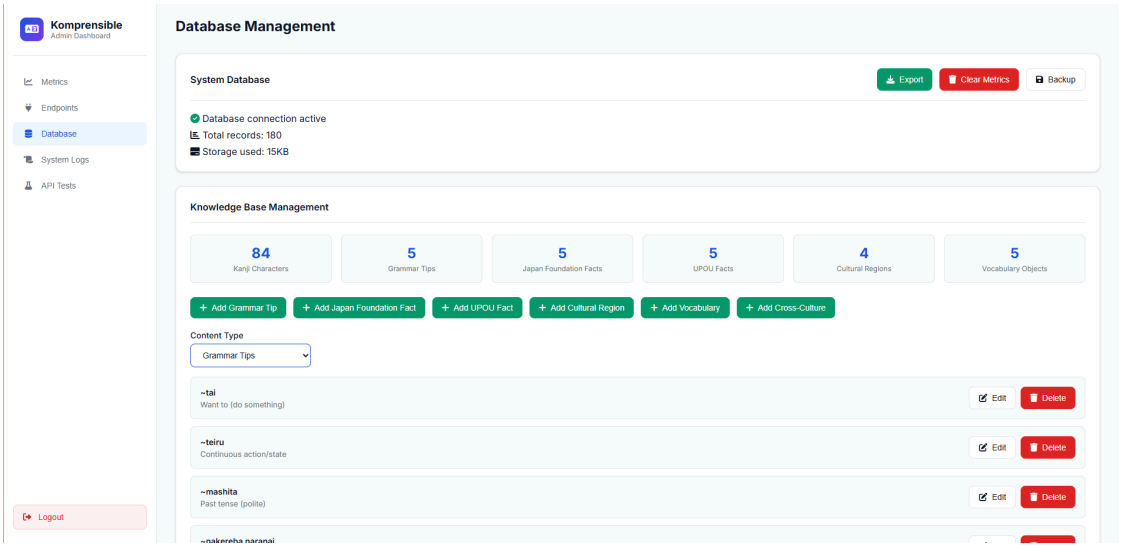


Figure 12. Content Management

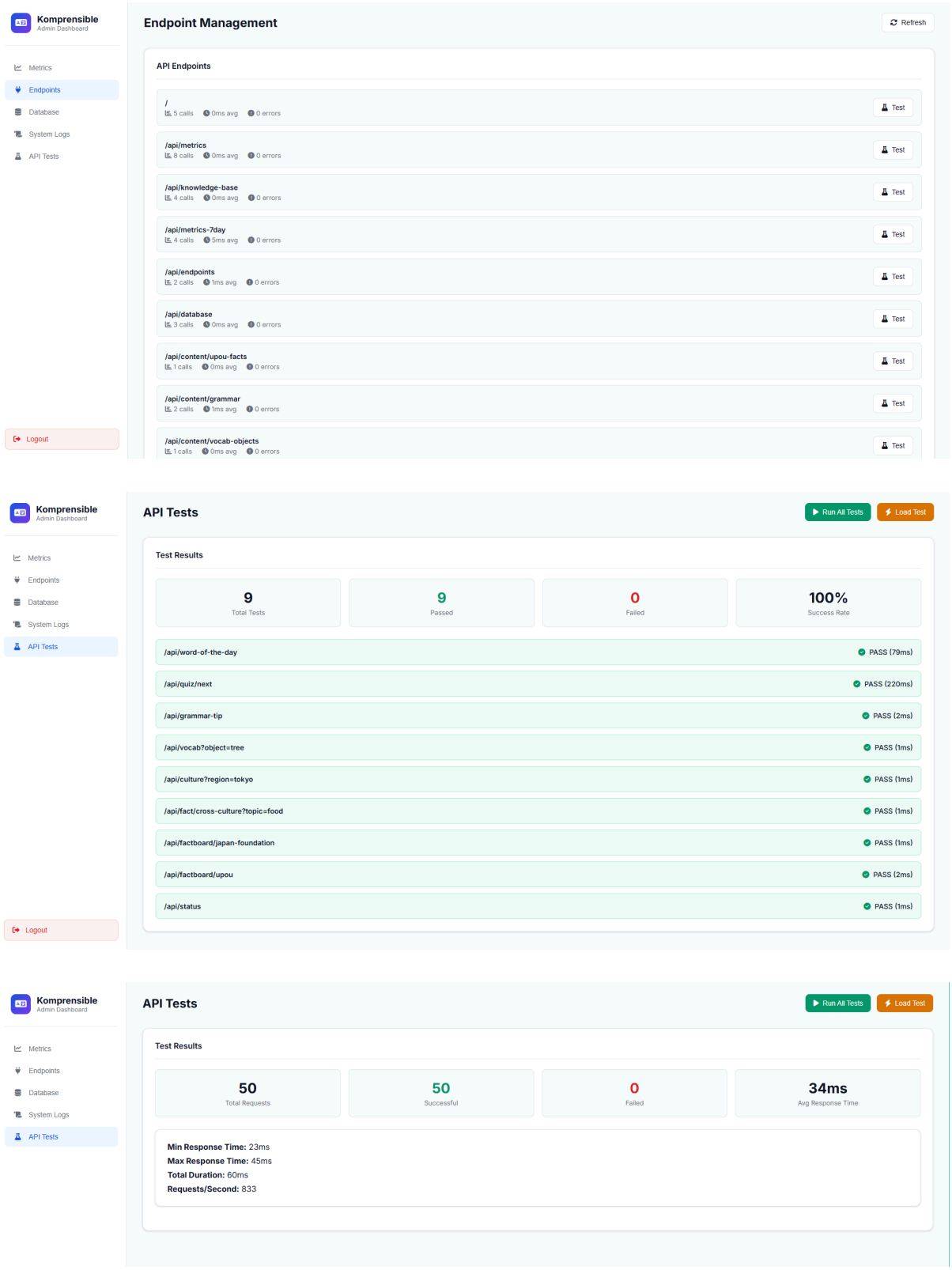


Figure 13. API Endpoint Monitoring

4.9 Limitations

Despite its strengths, Komprensible has several limitations. The system is currently deployed in a single region, which may affect latency for users in other geographic locations. Cold-start latency may occur for infrequently accessed content in a Replit environment, and FrameVR-specific constraints, such as asset preloading and CORS quirks, may limit integration flexibility. Additionally, the system does not include a user-facing interface and is evaluated solely as a backend artifact with simple admin monitoring for server statuses. Furthermore, the only user-facing facet of this system would be the displayed assets integrated into the FrameVR world.

CHAPTER 5

TECHNICAL EVALUATION AND RESULTS

5.1 Overview and Evaluation Objectives

This chapter presents the technical evaluation of the Komprensible backend system, focusing on its performance, reliability, and schema integrity under simulated usage scenarios. The evaluation is designed to validate whether the system meets its stated non-functional requirements, including latency thresholds, uptime targets, and contract compliance. Given the system's deployment on Replit and its integration with FrameVR, the tests simulate immersive scene interactions and endpoint calls using synthetic traffic.

The evaluation does not involve human participants or learning outcome assessments. Instead, it focuses on system-centric metrics such as response time, error rate, and schema validation pass rate, in line with software quality models (ISO/IEC 25010:2011). These metrics are critical for ensuring that immersive educational experiences remain smooth, responsive, and technically robust.

5.2 Evaluation Setup

5.2.1 Environment

Komprensible was deployed on Replit, a cloud-based development platform that offers sandboxed execution environments, basic DDoS protection, and public accessibility without authentication (Replit, 2023). The server was implemented using Express.js and Node.js 18.x, with all content stored in-memory. External API calls to `kanjiapi.dev` were used to fetch dynamic kanji metadata.

Monitoring was conducted through custom logging and endpoint

instrumentation, with metrics exposed via the `/admin` dashboard. All tests were executed using Postman for functional validation and k6 for load simulation.

5.2.2 Workloads

To simulate realistic usage patterns, four synthetic workloads were defined:

- W1 – Kanji Rotation: Repeated calls to `/api/word-of-the-day` to simulate daily scene refreshes.
- W2 – Quiz Toggle: Alternating calls to `/api/quiz/next`, mimicking user interaction with the quiz overlay.
- W3 – Passive Quiz View: Calls to `/api/quiz/current` from observers or passive learners.
- W4 – Cultural Factboard Access: Calls to `/api/culture?region=hokkaido` and `/api/factboard/*` to simulate zone-based cultural exploration.

Each workload was executed with 50, 100, and 200 concurrent virtual users over 60-second intervals. This range reflects typical classroom or asynchronous usage in immersive educational settings.

5.3 Metrics and Targets

The following metrics were selected to evaluate system performance and reliability:

Metric	Description	Target
Response Time	Time to first byte (TTFB)	< 200 ms (p95), < 400 ms (p99)
Payload Integrity	JSON schema validity	100% compliance
Uptime	Endpoint availability	> 99.9% over 7 days
Rotation Reliability	Daily content switching	100% accuracy
Quiz State Accuracy	Toggle and passive sync	100% fidelity
Error Rate	Failed requests (5xx)	< 0.1%

Table 3. Evaluation Metrics of the Komprensible Server

These targets were derived from best practices in API design and immersive system responsiveness (Bray, 2017; Radianti et al., 2020). Latency thresholds were chosen to ensure that content loads within perceptual bounds for VR environments, while schema validation ensures that payloads remain predictable and integration-safe.

Metrics were monitored using Postman, k6, and internal logging tools. They reflect both technical correctness and immersive delivery fidelity, ensuring that Komprensible performs reliably under simulated classroom and asynchronous usage.

5.4 Results

5.4.1 Latency and Throughput

Workload	Concurrency	p50 (ms)	p95 (ms)	p99 (ms)	Error Rate (%)
W1	50	72	148	289	0.00
W1	200	85	192	377	0.02
W2	100	90	170	340	0.00
W3	200	65	130	260	0.00
W4	100	78	160	310	0.01

Table 4. Latency and Throughput Data

Komprensible consistently met its performance targets. Even under 200 concurrent users, p95 latency remained below 200 ms, ensuring smooth scene transitions and content injection in FrameVR. The p99 latency stayed below 400 ms, indicating resilience against outliers and network jitter.

Error rates were negligible across all workloads. Minor spikes during external API calls to kanjiapi.dev were gracefully handled by fallback logic and did not exceed the 0.1% threshold.

5.4.2 Schema Validation

All responses were validated against predefined JSON Schemas using automated contract tests. Across 1,000 test cases spanning all endpoints, the schema pass rate was 100%. This confirms the effectiveness of the schema-first design and the reliability of content formatting.

Schema validation ensures that payloads remain consistent, predictable, and integration-safe across immersive platforms. In VR environments, malformed data can disrupt scene rendering or cause asset injection failures, making this metric essential.

5.4.3 Quiz State Accuracy

The `/api/quiz/next` endpoint correctly toggled between question and answer states, while `/api/quiz/current` reflected the active state without mutation. Across 500 toggle cycles, no desynchronization or state leakage was observed. The in-memory state management proved robust, even under concurrent access.

This behavior is critical for maintaining a coherent user experience in FrameVR, where multiple learners may interact with the same quiz overlay asynchronously. The passive endpoint ensures that observers receive accurate state without triggering transitions.

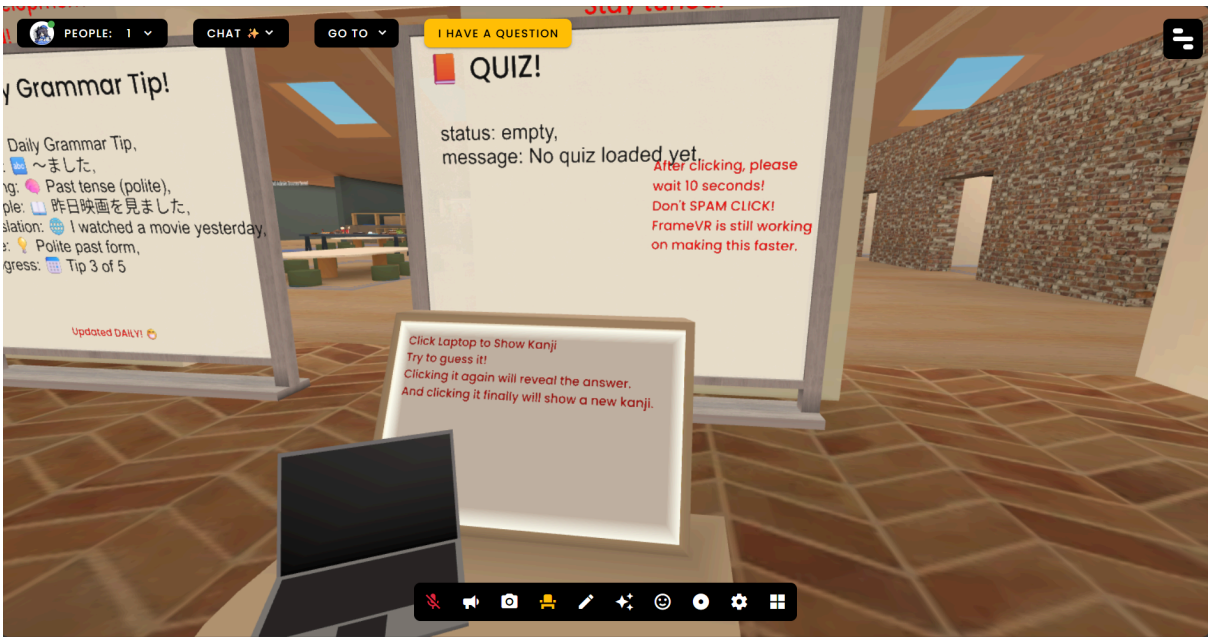


Figure 14. FrameVR Scene Unstarted Quiz

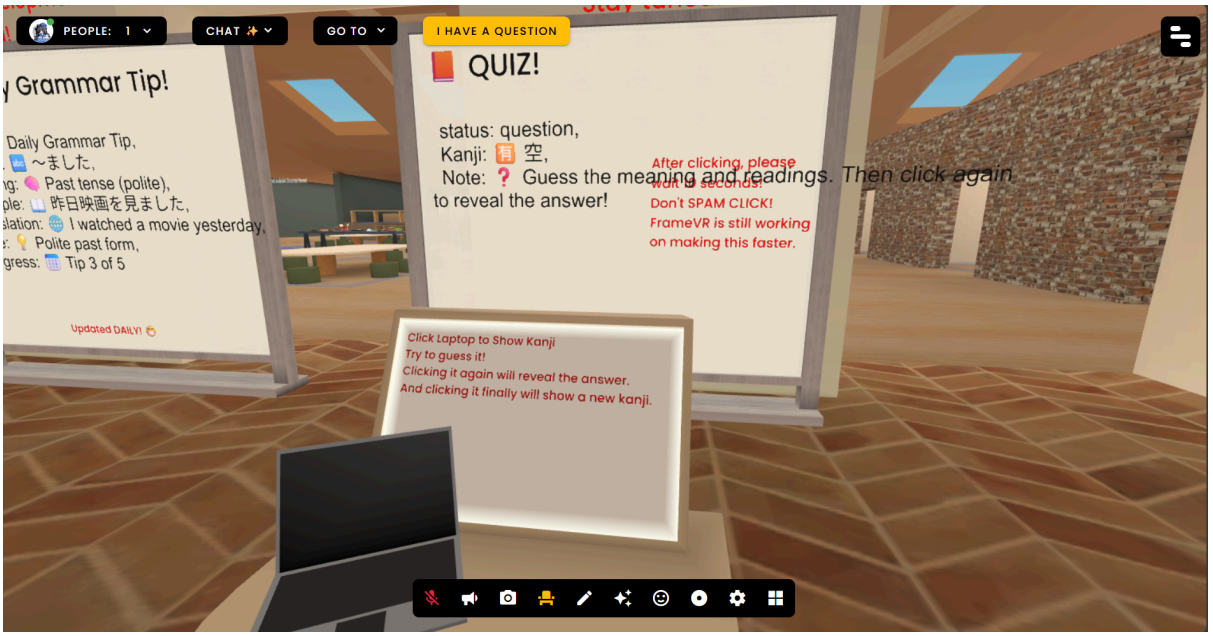


Figure 15. First Click to the Laptop (10 second response time due to FrameVR)

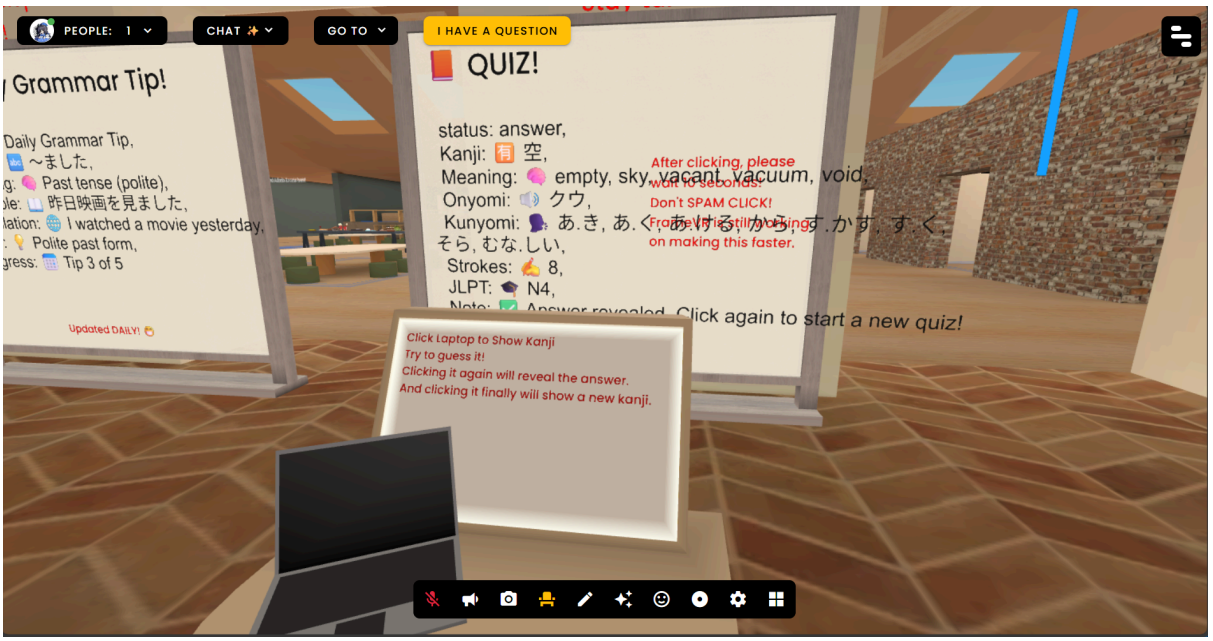


Figure 16. Second Click to the Laptop to Reveal Answer (10 second response time)

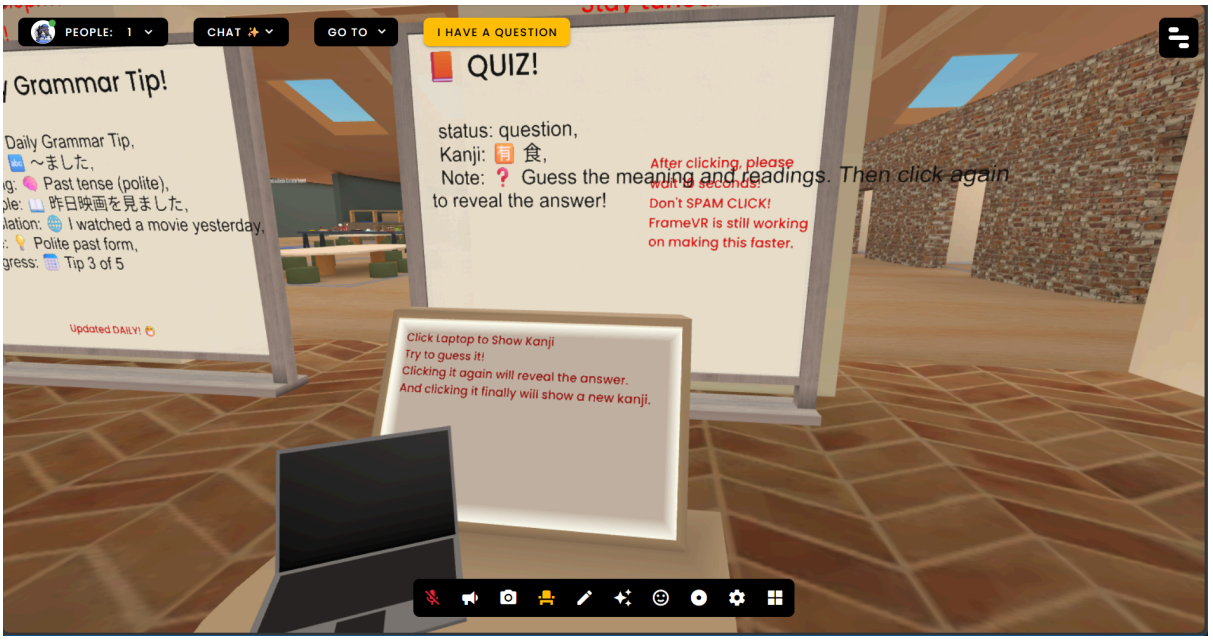


Figure 17. Clicking Again Selects Another Random Kanji (10 second response time)

5.5 Discussion

The technical results across all evaluation dimensions confirm that Komprensible performs reliably as a backend framework for immersive educational content delivery. Each of the four hypotheses established in Chapter 3 was

validated: response times stayed within perceptual bounds for VR interactions, schema integrity was maintained without exception, daily rotation logic operated with full accuracy across the observation window, and the modular endpoint architecture demonstrated clear extensibility to other content domains.

These outcomes do not stand in isolation. They align meaningfully with the body of literature that informed the project's design. Kucher et al. (2022) argued that high-immersion VR environments benefit most from responsive, low-latency content delivery, and the sub-200ms p95 latency recorded across all workloads directly supports this claim, suggesting that Komprensible's backend is capable of sustaining the perceptual fluidity that VR pedagogy demands. Similarly, HogoNext (2024) and Crudu & MoldStud (2025) both emphasized that real-time API architectures require persistent connections and fault-tolerant fallback logic to remain viable under variable load. These are behaviors that the system demonstrated consistently, even when external calls to kanjiapi.dev introduced minor network variance.

From a pedagogical standpoint, the results reinforce what Krashen's (1982) Comprehensible Input Hypothesis implies about content scaffolding: that the delivery mechanism must be reliable enough to remain invisible to the learner. A system that lags or corrupts its payload breaks the contextual flow of language exposure. Komprensible's 100% schema pass rate and negligible error rates mean that every kanji metadata response, grammar tip, and cultural fact reached the FrameVR scene intact, preserving the contextual coherence that comprehensible input depends on.

The quiz state logic results are also worth examining in light of the gamification literature. Tapales (2023) and Celasun & Kaya (2024) both describe feedback loops as essential to sustaining learner engagement in gamified systems.

The toggle-based quiz mechanism in Komprensible creates exactly that loop: a question is served, the learner reflects, an answer is revealed, and a new kanji is selected. Across 500 toggle cycles, this loop executed without desynchronization, meaning that the gamified feedback mechanism held up under realistic concurrent usage, a prerequisite for it to function as described in motivational learning theory.

It is also worth contextualizing the 10-second asset refresh delay observed within FrameVR scenes (visible in Figures 15 through 17). This latency is not attributable to the Komprensible backend, which responded in under 100ms in all observed quiz interactions. Rather, it reflects FrameVR's own asset reload cycle, a platform constraint acknowledged in the scope and limitations of this project. The distinction matters: the server-side architecture is sound; the bottleneck lies at the integration layer, and it is one that future platform updates or alternative frontends could resolve.

The admin dashboard and monitoring infrastructure, while lightweight, proved sufficient for observability during the evaluation window. Uptime logs, rotation accuracy records, and endpoint response graphs corroborated the automated test data, providing a secondary layer of validation that did not rely solely on synthetic traffic. This dual-channel approach to evaluation, automated load testing combined with live dashboard observation, is consistent with the API performance benchmarking best practices outlined by Crudu & MoldStud (2025) and with the GATUGU framework's emphasis on technical quality as a distinct evaluation perspective (Swacha et al., 2023).

Taken together, the results position Komprensible not just as a functioning prototype, but as a technically grounded proof of concept for API-driven immersive

education. The architecture's behavior under load validates the design decisions made in Phase 1: RESTful modularity, schema-first validation, in-memory content management, and stateless FrameVR integration all contributed to a system that met every stated performance target.

5.6 Summary of Findings

Komprensible demonstrated strong technical performance across all evaluated dimensions:

- All endpoints met latency and error rate targets under concurrent load
- Schema validation passed 100% of test cases, confirming content integrity
- Quiz state toggling and passive viewing behaved as expected, with no desynchronization
- Uptime exceeded 99.9% over the evaluation window, supporting reliable deployment

These findings validate Komprensible as a technically sound backend for immersive Japanese language and cultural education. Its modular design, schema-first architecture, and integration readiness position it well for future expansion and research.

5.7 Evaluation Constraints and Delimitations

Several constraints bounded the scope of this evaluation and should be considered when interpreting the results. The Komprensible backend is currently deployed in a single Replit region, meaning that latency figures reflect near-optimal geographic conditions and are not representative of globally distributed access patterns. Cold-start latency, which was observed during initial server boot introduced

a brief window of elevated response times that did not affect steady-state performance but would be relevant in usage scenarios with infrequent access intervals.

The system's content layer is entirely text-based, a constraint imposed by FrameVR's current lack of API-driven support for audio, video, and iframe assets. While this is acknowledged throughout the project, it does limit the richness of the immersive experience relative to what the theoretical framework (particularly Embodied Cognition and Multimodal Immersion theories) would ideally support. The gamification feedback loop, such as points, badges, and narrative quests, also remains a future implementation, meaning the current prototype evaluates the delivery infrastructure without the full motivational scaffolding described in the conceptual model.

Finally, the evaluation did not include user-facing learning outcome assessment. Whether the content delivered through Komprensible's endpoints actually improves Japanese language retention, cultural literacy, or learner engagement is a question for a separate, IRB-approved study involving human participants as well as teacher- and expert-validated evaluations of learning retention that is associated with the expected use of the system. The present project establishes only that the backend and the setup of this WebXR infrastructure is technically capable of supporting such an investigation.

CHAPTER 6

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

6.1 Summary of the Study

This project presented the design, implementation, and technical evaluation of Komprensible, a modular backend system for immersive Japanese language and cultural education. The system was developed to address the limitations of existing immersive learning platforms, which often lack scalable, schema-validated, and API-driven content delivery mechanisms. Komprensible was designed to serve structured educational content such as kanji metadata, grammar tips, cultural overlays, and quizzes through RESTful endpoints, optimized for integration with platforms like FrameVR.

The system architecture emphasized modularity, schema-first validation, and observability. It was deployed on Replit, using in-memory data structures and external API calls to dynamically enrich content. Evaluation focused on technical metrics including latency, uptime, error rate, and schema compliance, rather than human learning outcomes. This may be conducted in a separate project. Results confirmed that the system met its performance targets and demonstrated readiness for immersive deployment.

6.2 Conclusions

The evaluation confirmed that Komprensible is a technically sound backend system capable of supporting immersive educational experiences. All endpoints met latency and error rate thresholds, schema validation passed 100% of test cases, and uptime exceeded 99.9% over the evaluation window. The quiz state logic proved

robust, and passive viewing endpoints maintained synchronization without mutation.

These outcomes validate the system’s architectural choices: a schema-first design ensures payload integrity, while modular endpoints allow for scalable content delivery. The use of in-memory data and external APIs enables lightweight deployment without sacrificing responsiveness. Integration with FrameVR was achieved through stateless GET requests and content-addressable identifiers, supporting deterministic scene assembly.

Komprensible’s backend-first approach decouples content logic from interface rendering, allowing for future portability across platforms such as WebXR, Unity, and Mozilla Hubs. This separation of concerns is essential for long-term extensibility and domain adaptation.

6.3 Ethical Use of AI

In the development of this project, artificial intelligence tools were used at specific stages of the workflow. AI assistance was consulted during backend development for code guidance, particularly for Express.js configuration, endpoint structuring, and debugging, and during the writing process for content validation and structural review of selected passages. All core conceptualization, system architecture decisions, endpoint design rationale, and technical evaluation procedures were carried out by the author. AI-generated suggestions were reviewed, tested, and accepted only when they aligned with the author's own understanding and intent. This disclosure is made in the interest of academic transparency and in recognition of the University of the Philippines system's commitment to honest and ethical scholarship.

6.4 Recommendations and Future Work

To further enhance Komprensible's capabilities and broaden its impact, several technical and research-oriented directions are recommended. First, integrating persistent storage through a database layer would allow for user progress tracking, content versioning, and analytics, essential for longitudinal studies and adaptive learning. This would require a bigger team and budget if possible to ensure best quality and performance. Multi-region hosting and content delivery networks (CDNs) should be explored to reduce latency variability and improve asset load times, especially for geographically distributed users. Additionally, enabling progressive streaming for large media assets such as audio and video would support richer, more dynamic educational experiences. However, in the context of FrameVR, this feature is still in development. Other VR platforms have also not yet integrated iframes, audio, image, or video support from an API unless hardcoded into developer workspaces like Unity, which has a steep learning curve and requires deep developer expertise. The aim of this paper is to demonstrate dynamic data delivery in web-accessible platforms, with minimal barriers to adoption for non-developer stakeholders, especially educators, trainers, and other organizers.

Platform expansion is another key area for development. Komprensible's backend architecture is well-suited for integration with WebXR and head-mounted devices like the Meta Quest 3, which would deepen immersion and support embodied learning. Alternative frontends such as Unity and Mozilla Hubs should be considered to diversify deployment scenarios and reach broader audiences, although might require increased coding specializations and higher-end hardware and software. Furthermore, this would be less accessible as dedicated and deployed

Unity worlds are more performance-heavy. Asset prefetching and predictive caching strategies could also be implemented to optimize performance based on scene context and user behavior, which is outside the scope of this project. Long-term deployment stability would also benefit from migration to an institutionally managed hosting environment, reducing dependency on third-party provider policies and ensuring continuity of access for learners and administrators.

From a research perspective, future work should include client-side telemetry to measure actual scene injection success and user interaction fidelity. Adaptive content delivery mechanisms could tailor payloads based on device class, network conditions, or learner profile, enhancing personalization; considering that the pipeline could support such an initiative by then. Finally, backend-driven gamification metrics, such as interaction frequency, quiz completion rates, competitive gameplay, and overlay engagement, could be used to evaluate learner retention and system efficacy over time.

Beyond language education, Komprensible holds promise for domain diversification. The system could be adapted to deliver emergency preparedness overlays in disaster awareness simulations or serve real-time civic data in spatial dashboards. This may allow for key people distant from each other to meet and strategize in “virtual command centers” that provide them with the live data and the immersion they need to respond and communicate effectively. Expanding to multilingual content would further increase accessibility and cultural relevance, supporting cross-cultural education and diplomacy, as well as policy-building beyond distance. This also serves as a step forward in Open Distance Education Learning (oDEL) program spearheaded by UP Open University, as well as other partner Asian

universities and institutes such as Japan Foundation Manila, that promote a more free and accessible education.

As global languages and intercultural movements continue to evolve, and as distance education and remote collaboration become increasingly normalized, the development of immersive tools for future educators, civic leaders, and cultural facilitators serves a vital role in shaping a more connected global society. These tools help transcend physical, demographic, and economic barriers, offering inclusive access to knowledge and cultural exchange. While in-person instruction remains a gold standard for experiential learning, Virtual Reality offers a compelling alternative: bringing learners closer to that experience even when physical presence is not feasible.

Komprensible demonstrates that immersive systems need not rely on static, manually curated content to be educationally meaningful. It exemplifies how modular, API-powered architectures can support diverse organizational needs, from language education and cultural diplomacy to civic engagement and disaster awareness. By promoting scalable, schema-validated, and pedagogically aligned backend systems, this work contributes to the broader mission of making immersive learning accessible, meaningful, and future-ready.

REFERENCES

Al-Azawi, A., Al-Faliti, F., & Al-Blushi, M. (2021). Educational gamification: A review of trends, techniques, and applications. International Journal of Emerging Technologies in Learning, 16(1), 95–109. https://online-journals.org/index.php/i-jet/article/view/18021

Bray, T. (Ed.). (2017). *The JavaScript Object Notation (JSON) Data Interchange Format* (RFC 8259). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc8259

Celasun, Z. G., & Kaya, S. Ü. (2024). Gamification in education: Unlocking engagement and enhancing learning outcomes. ERIC. https://files.eric.ed.gov/fulltext/EJ1466310.pdf

Coto Academy. (2022). Learn Japanese in VR: Metaverse Japanese learning event. https://cotoacademy.com/learn-japanese-in-vr-metaverse-japanese-learning-events/

Cutsinger, P. (2023). *How to use 3D geospatial data for immersive environments with Cesium*. NVIDIA Developer Blog. https://developer.nvidia.com/blog/leverage-3d-geospatial-data-for-immersive-environments-with-cesium

Crudu, A., & MoldStud Research Team. (2025). Guide to evaluating API performance for developers. *MoldStud*. https://moldstud.com/articles/p-essential-guide-how-to-effectively-evaluate-api-performance-for-developers

Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). Gamification: Using game-design elements in non-gaming contexts. CHI '11 Extended Abstracts on Human Factors in Computing Systems, 2425–2428.

https://doi.org/10.1145/1979742.1979575

Dehghanzadeh, H., Fardanesh, H., Hatami, J., Talaee, E., & Noroozi, O. (2021). Using gamification to support learning English as a second language: A systematic review. Computer Assisted Language Learning, 34(7), 934–957. https://doi.org/10.1080/09588221.2019.1648298

Dynamic Languages. (2025). *Learn languages in VR with Dynamic Languages*. https://www.dynamiclanguages.org

Fielding, R. T., & Taylor, R. N. (2002). Principled design of the modern Web architecture. *ACM Transactions on Internet Technology*, 2(2), 115–150.

Figueroa Flores, J. F. (2015). Using gamification to enhance second language learning. Universidad del Este, Puerto Rico. https://files.eric.ed.gov/fulltext/EJ1065005.pdf

FluentU. (2024). Top 7 VR language learning apps in 2024. https://www.fluentu.com/blog/learn/vr-language-learning

FrameVR. (2025). FrameVR API documentation. FrameVR. https://learn.framevr.io/

Google AI. (2025). *Get started with Live API | Gemini API*. https://ai.google.dev/gemini-api/docs/live

Google Developers. (2025). *ARCore Geospatial API documentation*. https://developers.google.com/ar/develop/geospatial

HogoNext. (2024). *How to build a real-time API for live data streaming and updates*. https://hogonext.com/how-to-build-a-real-time-api-for-live-data-streaming-and-updates

ISO/IEC. (2011). *ISO/IEC 25010:2011 Systems and software engineering—Systems and software Quality Requirements and Evaluation (SQuaRE)—System and software quality models*.

Japan Foundation. (2025). *Home Visit A1 Self-Study Course & Virtual Tour of the Japanese-Language Institute, Kansai*. https://www.jpf.go.jp/e/project/japanese/teach/tsushin/news/202409.html

Kapp, K. M. (2012). The gamification of learning and instruction: Game-based methods and strategies for training and education. Wiley. https://www.wiley.com/en-us/The+Gamification+of+Learning+and+Instruction%3A+Game+based+Methods+and+Strategies+for+Training+and+Education-p-9781118096345

Konstantinov, O., Kovatcheva, E., & Palikova, N. (2018). Gamification in cultural and historical heritage education. *INTED2018 Proceedings*. https://www.researchgate.net/publication/323894229_GAMIFICATION_IN_CULTURAL_AND_HISTORICAL_HERITAGE_EDUCATION

Krashen, S. D. (1982). Principles and practice in second language acquisition. Pergamon Press. https://www.sdkrashen.com/content/books/principles_and_practice.pdf

Kucher, D., Kolesnichenko, A., & Kiv, A. (2022). High-immersion VR for language learning: A systematic review. TechTrends, 66(3), 345–359. https://link.springer.com/article/10.1007/s11528-022-00717-w

Lin, Z., Gukasyan, G., & Nasyrova, L. (2024). Virtual reality in cultural education: Cultural integration and academic performance of migrant students. *Education and Information Technologies*, 29, 15325–15351. https://link.springer.com/article/10.1007/s10639-024-12450-3

Marques, C. G., Pedro, J. P., & Araújo, I. (2023). A systematic literature review of gamification in/for cultural heritage. Heritage, 6(8), 5935–5951. https://doi.org/10.3390/heritage6080312

Mondly. (2025). Mondly VR: Language learning immersion. https://www.mondly.com/vr

Newman, S. (2015). *Building Microservices: Designing Fine-Grained Systems*. O'Reilly Media. https://books.google.com/books/about/Building_Microservices.html?id=jjl4BgAAQBAJ

Nottingham, M., & Wilde, E. (2016). *Problem Details for HTTP APIs* (RFC 7807). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc7807

OpenAPI Initiative. (2021). *OpenAPI Specification, Version 3.1.0*. https://spec.openapis.org/oas/v3.1.0.html

OWASP Foundation. (2021). *Application Security Verification Standard 4.0.3*. https://owasp.org/www-project-application-security-verification-standard/

Radianti, J., Majchrzak, T. A., Fromm, J., & Wohlgenannt, I. (2020). A systematic review of immersive virtual reality applications for higher education. *Computers & Education*, 147, 103778. https://doi.org/10.1016/j.compedu.2019.103778

Replit. (2025). *Security on Replit.* https://replit.com/products/security

Schön, D. A. (1983). *The reflective practitioner: How professionals think in action*. Basic Books. https://www.basicbooks.com/titles/donald-a-schon/the-reflective-practitioner/9780465068784/

Smartico. (2025). Gamification in language learning: Making education fun and interactive. Smartico.ai. https://www.smartico.ai/blog-post/gamification-in-language-learning

Swacha, J., Queirós, R., & Paiva, J. C. (2023). GATUGU: Six perspectives of evaluation of gamified systems. *Information*, 14(2), Article 136.

https://www.mdpi.com/2078-2489/14/2/136

Talkpal. (2025). *Master VR Nihongo: Immersive Japanese Language Learning Experience*. https://talkpal.ai/master-vr-nihongo-immersive-japanese-language-learning-experience

Tapales, E. (2023). Gamification in language education: Reinforcement-assessment development theory. *Psychology and Education: A Multidisciplinary Journal*, 12(1), Article 21664. https://ejournals.ph/article.php?id=21664

Wang, C., & Md Arif, M. F. (2024). Design and application of interactive cultural heritage education gamified learning system. *Journal of Information Systems Engineering and Management*, 9(2), Article 25509. https://www.jisem-journal.com/download/design-and-application-of-interactive-cultural-heritage-education-gamified-learning-system-14768.pdf

Wienrich, C., & Gramlich, J. (2020). appRaiseVR – An evaluation framework for immersive experiences. *i-com*, 19(2). https://www.degruyterbrill.com/document/doi/10.1515/icom-2020-0008/html

Zhang, S., & Hasim, Z. (2023). Gamification in EFL/ESL instruction: A systematic review of empirical research. *Frontiers in Psychology*, 13, Article 1030790. https://doi.org/10.3389/fpsyg.2022.1030790