MravalJam - Interactive Musical Setting Vol.2

Giorgi Maglakelidze - a61475

Davit Chikobava - a61476

Thesis presented to the School of Technology and Management in the scope of the Bachelor in Informatics.

Supervisors:

Prof. Rui Pedro Lopes

Ahmed Gamal Ibrahim

This document does not include the suggestions made by the board.

Bragança

2024-2025

MravalJam - Interactive Musical Setting Vol.2

Giorgi Maglakelidze - a61475

Davit Chikobava - a61476

Thesis presented to the School of Technology and Management in the scope of the Bachelor in Informatics.

Supervisors:

Prof. Rui Pedro Lopes

Ahmed Gamal Ibrahim

This document does not include the suggestions made by the board.

Bragança

2024-2025

Dedication

To our music teachers: Irodi Tikaradze and Kote Jaiani. And to our target audience, middle school student: Lazare Kezherashvili.

-Giorgi, Davit

Abstract

The MravalJam project explores the potential of gamified music education to address the challenges of engaging school students in learning more about music. The platform is built using the Unity Game Development Environment and introduces students to foundational musical concepts through an interactive sticker-based system. Each sticker represents a sound layer, allowing students to create melodies while learning about the interplay of musical elements. To foster cultural appreciation, the platform features themed levels highlighting diverse musical traditions. These themes are designed to expose students to global musical diversity while emphasizing creativity and engagement. The application integrates gamification principles, including progress tracking and immediate feedback, to enhance motivation and learning outcomes. By transforming music education into an accessible and enjoyable experience, MravalJam aims to bridge the gap between traditional music curricula and the interests of modern students, leveraging technology to create a more dynamic and inclusive educational environment.

Contents

List of Figures

Acronyms

DAW Digital Audio Workstation.

ESTiG Escola Superior de Tecnologia e Gestão.

IPB Instituto Politécnico de Bragança.

JSON JavaScript Object Notation.

MIDI Musical Instrument Digital Interface.

SeMI Interactive Musical Setting with Deep Learning and Object Recognition.

UI User Interface.

YOLO You Only Look Once.

Chapter 1

Introduction

Music education has long been regarded as a vital component of cultural and creative development. However, in many educational systems worldwide, it has often been relegated to a secondary status, perceived as unengaging or even unnecessary by students and educators alike. Researches [1] [2] highlight a significant disjuncture between the declining role of music education in schools and the importance of music in youth culture and the modern knowledge economy. This gap is driven by outdated curricula, lack of resources, and an overemphasis on standardized evaluations, leaving music education unable to inspire or engage students effectively.

The value of music education, however, extends far beyond its role in cultural development. Music education has been shown to improve perception and cognition, enhancing a child's ability to process auditory and visual information, and contributing to overall academic achievement. Furthermore, music education supports emotional development, helping children express and regulate their emotions, while also promoting social skills [2].

Importantly, music education has been found to benefit at-risk and special needs students by providing a medium for self-expression and engagement that traditional academic settings often fail to offer. Despite this extensive evidence, music education remains undervalued in many school systems, particularly in developing countries like Georgia, where it is often reduced to uninspiring lectures or neglected altogether.

In Georgia, the challenges in music education are acutely felt. School students are rarely exposed to the richness of music or the diversity of global musical traditions. Recognizing these issues and inspired by personal experiences, development team sought to reimagine how music education can be delivered to young students in a way that is engaging, interactive, and culturally enriching.

MravalJam, the project presented in this paper, is a gamified music education platform designed to address these challenges. Built using the Unity Game Development Environment [3], *MravalJam* introduces students to foundational music theory concepts such as harmony, rhythm, and melody through an interactive sticker-based system. Each level of the game is themed to highlight a specific musical style or tradition, fostering both musical and cultural appreciation. By leveraging gamification principles, the platform aims to make music education accessible and enjoyable for primary to middle school students, combining creativity, interactivity, and systematic learning.

MravalJam integrates interactive gameplay mechanics that support both cognitive and emotional development. By exposing students to diverse musical traditions, the platform encourages cultural awareness and empathy, helping nurture well-rounded, creative, and socially aware individuals. This project seeks not only to address the current shortcomings in music education but also to highlight its potential as a transformative force in child development.

Chapter 2

State-of-the-art

2.1 Building on Prior Works

One of the significant inspirations for *MravalJam* is the project titled Interactive Musical Setting with Deep Learning and Object Recognition (SeMI) by Cardoso and Lopes (2020) [4]. This project explored the integration of machine learning, digital image recognition, and sound generation to create an interactive musical learning environment.

The original *SeMI* application utilized image recognition powered by You Only Look Once (YOLO) [5] to detect physical objects in real-time. Each recognized object was associated with a pre-defined musical texture, enabling the creation of dynamic soundscapes. This system allowed students to experiment with designing sound textures and explore concepts of music composition in an engaging and intuitive way.

SeMI in its initial form was highly innovative, providing students with a unique opportunity to connect physical objects with corresponding sound outputs. However, its reliance on image recognition introduced complexities that limited its accessibility for younger audiences and middle school students.

Modifications in the Current Project

MravalJam builds upon the foundational ideas of the original project while making significant modifications to better align with the goals of teaching music theory to primary to middle school students:

- **Simplification**: The image recognition component was removed to simplify the interaction model, not to be limited by real-life objects around the user, and to reduce the technological learning curve for students.
- **Focus on Thematic Levels**: Instead of object recognition, our version uses a sticker-based system where students create sound layers by combining thematic stickers on a virtual board.
- **Introduction of Musical Styles**: *MravalJam* introduces themed levels (e.g., space, Middle Eastern) to expose students to various musical styles while teaching harmony, rhythm, and melody.

By adapting the original concept to focus on interactive learning through gamification and cultural themes, this project aims to make music education more engaging and accessible, especially for younger learners.

2.2 Introduction to Music Education Tools

Music education tools and games have become increasingly important in making music theory accessible and engaging, particularly for younger audiences. With the rise of digital technology, tools such as *Yousician*[6], *Rhythm Cat*[7], *MuseScore*[8], *Audacity*[9], *GarageBand*[10], and *Soundtrap*[11] have gained prominence for their innovative approaches to teaching music concepts. These tools highlight the potential of gamified and digital learning to enhance music education, offering interactive experiences that cater to diverse learning needs.

Digital tools have been increasingly integrated into modern music education, as noted in the study *"Digital Tools of Universal Music Education"* [12]. These tools combine multimedia elements like graphics, audio, animations, and interactivity to create rich and engaging learning environments. They enable students to actively participate in the learning process, developing creativity and cognitive skills in ways that traditional methods often cannot. Interactive displays, music notation editors, and audio software applications allow teachers to provide tailored and effective learning experiences for students with diverse needs.

This subsection provides an overview of six popular tools—*Yousician*, *Rhythm Cat*, *MuseScore*, *Audacity*, *GarageBand*, and *Soundtrap*—that significantly influenced the development of *MravalJam*. By examining their strengths and limitations, opportunities to create a more comprehensive and engaging platform for primary to middle school students were identified. Insights from relevant studies further underscore the role of these tools in reshaping the music education landscape.

Yousician

Yousician [6] is a music education platform designed to teach instruments and music theory through gamified lessons. It uses real-time feedback to guide users as they play guitar, piano, ukulele, and other instruments, helping them correct mistakes and improve accuracy. Its structured lessons and achievement-based progression system have made it a popular choice for learners of all ages.

Yousician excels in its ability to motivate learners through gamified features, such as progress tracking, performance rewards, and interactive feedback. These elements are key to maintaining engagement, especially for younger users who thrive on positive reinforcement and clear goals. However, *Yousician*'s emphasis on individual instrument performance leaves a gap in exploring broader musical concepts such as harmony and sound layering, which are essential for developing a holistic understanding of music.

For the development of *MravalJam*, *Yousician* provided inspiration in its use of real-time feedback and structured lessons. However, recognizing the limitations of a narrow

focus on instrument-specific training, *MravalJam* focuses on introducing thematic lessons that incorporate cultural diversity and foundational music theory. By integrating these elements into an engaging gamified environment, *MravalJam* addresses a broader range of educational goals.

Rhythm Cat

Rhythm Cat [7] is an interactive rhythm game aimed at developing a sense of timing and beat recognition in learners. The game features simple visuals and mechanics, where users tap along to rhythms displayed on the screen. Its simplicity and engaging design make it particularly effective for young learners starting their musical journey.

The primary strength of *Rhythm Cat* lies in its accessible interface and focus on rhythm training. By providing an intuitive and visually appealing experience, the game encourages students to practice rhythm exercises without feeling overwhelmed. However, its limited scope—focused exclusively on rhythm—does not address other critical aspects of music education, such as melody, harmony, and sound composition.

In designing *MravalJam*, the accessibility and intuitive gameplay of *Rhythm Cat* served as key inspirations. *MravalJam* builds on these strengths by creating a sticker-based system where students experiment with sound layering in a non-intimidating way. Additionally, by incorporating a broader range of musical elements, *MravalJam* fosters a more comprehensive learning experience that encourages creativity and exploration.

MuseScore

MuseScore [8] is a widely used digital tool for music notation and composition. It enables students to compose, arrange, and play back their creations, making it an invaluable resource for teaching music theory and composition in a hands-on manner. As highlighted in *"Digital Tools of Universal Music Education"* [12], *MuseScore* allows users to input music through a mouse, keyboard, or *Musical Instrument Digital Interface (MIDI)* system and offers features like integrated playback, text annotations, and dynamic markings.

These features empower students to experiment with musical ideas and gain immediate auditory feedback.

For the development of *MravalJam*, the accessibility and functionality of *MuseScore* influenced the design of the sticker-based system, where students can layer sounds and experiment with musical textures. The emphasis on creativity and experimentation aligns with *MravalJam*'s goals of fostering an intuitive and interactive learning environment.

Audacity

Audacity [9] is a powerful open-source audio editor widely used for sound recording and manipulation. It supports multiple audio formats and offers advanced editing features like cutting, mixing, and applying effects such as echo and noise reduction. *Audacity*'s simplicity and versatility make it an excellent tool for students to explore sound design and develop technical skills related to audio production.

In *MravalJam*, the concept of enabling students to interact with audio elements drew inspiration from tools like *Audacity*, where users can modify and layer sounds. Although *MravalJam* simplifies the process for younger audiences, the underlying principle of empowering students to engage with sound creatively remains central to its design.

GarageBand

GarageBand [10] is a *Digital Audio Workstation (DAW)* developed by Apple that has become a popular tool for music creation and education. Its intuitive interface and extensive library of virtual instruments, loops, and sound effects make it an accessible platform for beginners and advanced users alike. *GarageBand* allows students to compose, edit, and produce music in an interactive and hands-on manner, providing an engaging way to learn music theory and composition.

One of the key advantages of *GarageBand* is its ability to simulate real instruments, enabling students without access to physical instruments to experiment with musical ideas. The platform also supports multitrack recording, allowing students to layer sounds and create complex compositions. These features align closely with *MravalJam*'s focus

on sound layering and composition, serving as a source of inspiration for its sticker-based system. However, unlike *GarageBand*, which targets a wide range of users, *MravalJam* simplifies the interface to cater specifically to primary and middle school students.

Soundtrap

Soundtrap [11] is an online, collaborative digital audio workstation that allows users to create music in real-time. Its cloud-based nature makes it highly accessible, enabling students to work on projects from any location with internet access. The platform supports collaborative learning, where multiple users can simultaneously edit and produce music, fostering teamwork and communication skills.

The versatility of *Soundtrap* is evident in its wide range of features, including virtual instruments, audio recording, and *MIDI* support. Students can experiment with loops, effects, and multitrack editing to create unique compositions. This collaborative aspect inspired *MravalJam* to incorporate mechanics that encourage interaction and exploration, albeit in a single-user environment tailored for younger audiences.

2.3 Unity and Game Development in Education

Game development has emerged as a powerful medium for engaging students in various fields, including computer science, mathematics, and even music education. The Unity Game Development Environment, with its professional-grade features and ease of use, has become a widely adopted tool for creating educational experiences. Unity's flexibility allows developers to create immersive and interactive applications, making it an ideal platform for gamification in education. By combining technical precision with creative freedom, Unity has enabled the development of diverse educational tools that cater to a wide range of disciplines and learning styles.

The growing popularity of game-based learning highlights its effectiveness in fostering engagement and improving retention. Gamification, as a strategy, incorporates game elements like rewards, challenges, and interactivity into non-game contexts to enhance

the learning process. Unity's built-in functionalities, including its robust physics engine, versatile scripting environment, and extensive asset library, make it particularly suited for implementing gamified educational platforms. This section explores studies and projects that highlight the use of Unity and game development in educational contexts, providing valuable insights that shaped the development of *MravalJam*.

The study titled *"Engaging Students in Computer Science Education through Game Development with Unity"* [13] investigates the effectiveness of using Unity as an educational tool for school students. The researchers adopted a Design-Based Research approach to teach game development, combining tutorials, teacher explanations, and peer collaboration to create an engaging and motivational learning environment.

The study's findings emphasize the motivational power of game development to engage students in complex subjects like computer science. Students were encouraged to solve problems creatively and collaboratively, gaining technical skills while enjoying the process. This aligns with the goals of *MravalJam*, where Unity serves as the backbone for implementing a gamified music education platform. By leveraging Unity's robust game development capabilities, *MravalJam* combines interactivity, creativity, and systematic learning to provide an engaging educational experience. However, unlike the study, which focuses on teaching coding concepts, *MravalJam* targets younger audiences with no prior technical experience. The project emphasizes music theory and sound layering through an intuitive sticker-based system, ensuring accessibility for primary to middle school students. By adapting Unity's features to meet the specific needs of music education, *MravalJam* demonstrates the versatility of game development tools in diverse learning contexts.

Clear example of using games for educational purposes is *Minecraft: Education Edition* [14]. It is a specialized version of the popular sandbox game *Minecraft* [15], designed specifically for educational purposes. It provides students with a creative and interactive environment to explore topics such as mathematics, science, history, and programming through hands-on activities and challenges.

The platform's open-ended nature allows educators to design customized lessons and

collaborative tasks that encourage teamwork and problem-solving. For example, students can use *Minecraft: Education Edition* to simulate historical events, build models of molecular structures, or learn basic programming concepts using the in-game code builder. These features have made it a versatile tool for classroom learning across various disciplines.

While *Minecraft: Education Edition* focuses on a wide range of subjects, its approach to gamified learning aligns closely with the objectives of *MravalJam*. Both platforms aim to make learning engaging and accessible for younger audiences by integrating educational content into interactive experiences. However, *MravalJam* distinguishes itself by focusing specifically on music education. It offers structured lessons on harmony, rhythm, and melody through themed levels that introduce students to various musical styles, fostering both creativity and cultural appreciation.

The integration of gamification into educational tools has transformed traditional learning paradigms, making them more interactive and engaging. As discussed in the researched studies [12] [2], gamification incorporates elements such as rewards, challenges, leaderboards, and progress tracking to motivate learners. Unity's capabilities make it a powerful platform for implementing these features, as seen in projects like *MravalJam*, where gamification drives student engagement and fosters deeper learning.

In *MravalJam*, gamification elements are embedded throughout the gameplay. Students earn stars by completing levels and filling key stats such as harmony, melody, and rhythm. Additional challenges, like achieving badges in certain levels, provide further motivation and a sense of accomplishment. These elements are designed to keep students engaged while reinforcing their understanding of music theory. Unlike traditional methods that rely on rote memorization, gamification encourages active participation and experimentation, making the learning process enjoyable and effective.

Gamification also fosters collaborative learning by encouraging students to seek help from teachers when they encounter difficulties. For instance, the hint system in *MravalJam* provides subtle guidance without solving the level entirely, promoting teacher-student interaction. This approach not only strengthens the teacher's role but also helps identify

areas where students need additional support.

On top of exploring role of gamification in musical education and child development, it is crucial not to overlook importance of using Unity as an engine for game development. Unity's advanced features have played a deciding part in shaping the development of *MravalJam*. It enables interactions between stickers and the virtual board, while its scripting environment facilitates the implementation of dynamic gameplay mechanics such as sticker shuffling and sound synchronization. Additionally, Unity's Asset Store provides access to a vast library of resources, allowing developers to create visually appealing and immersive experiences.

In *MravalJam*, Unity's scene management tools ensure smooth transitions between levels, while its audio capabilities support the seamless integration of sound layers. These technical solutions align with the project's educational objectives, making Unity an indispensable tool for the development of gamified learning platforms. Thanks to that, *MravalJam* combines interactivity, gamification, and structured content to make music education engaging and accessible for primary to middle school students. The incorporation of gamification elements further enhances the learning experience, providing students with a rewarding and enjoyable way to explore music theory and sound composition.

2.4 Cultural Representation in Music Education

The integration of cultural representation into music education is essential for fostering an appreciation of the diverse traditions and values that shape the global musical landscape. The article *"Re-appraising Ideas of Musicality in Intercultural Contexts of Music Education"* [16] explores the importance of incorporating intercultural perspectives into music education. It critiques traditional music curricula for their limitations in embracing a pluralistic view of music, arguing that a truly intercultural music education must integrate diverse performing and learning practices alongside the beliefs and values of various musical systems. This call for an intercultural model resonates deeply with the objectives of *MravalJam*, which seeks to expose students to a wide range of musical traditions

through gamified, thematic levels.

A core concept highlighted in the article is *intermusicality*, which refers to the interconnectedness of musical traditions across different cultures. This idea is at the heart of *MravalJam*'s design, where themed levels, such as Middle Eastern music and space-based funk music, introduce students to distinct musical styles. By layering sounds and experimenting with musical textures, students develop an intuitive understanding of the diverse elements that contribute to harmony, melody, and rhythm in various cultural contexts. The project's approach aligns with the article's advocacy for breaking away from singular cultural lenses and fostering an appreciation of music as a universal language.

Building on these insights, the study *"Music Education and Cultural Identity"* [17] further emphasizes the relationship between music education and cultural identity. It highlights the dual influence of globalized education, which promotes a multicultural ethic of diversity, and the particularism of ethnic and cultural traditions in shaping musical awareness. This duality is reflected in *MravalJam*, which aims to balance the exposure to global musical traditions with an emphasis on the cultural and historical significance of each style. For example, the Middle Eastern-themed level not only introduces students to the distinctive sounds of the region but also encourages them to reflect on its historical and cultural roots, fostering both cognitive and emotional connections to the music.

Furthermore, the study discusses how music education serves as a platform for students to explore their own cultural identities while engaging with those of others. This aligns with the broader goal of *MravalJam*, which integrates cultural themes to instill a sense of global citizenship in students. By allowing students to experiment with culturally significant musical elements, the project encourages creativity and self-expression while promoting cross-cultural empathy and understanding.

The ethnomusicological perspectives discussed in [16] also emphasize the pedagogical value of incorporating diverse cultural practices into music education. *MravalJam* reflects these principles by creating interactive and immersive levels where students can explore different musical traditions in a hands-on manner. The interactive gameplay encourages students to experiment with sound layers associated with specific cultural themes, bridging

theoretical concepts with practical application. This method not only enhances their understanding of music theory but also deepens their appreciation for the cultural contexts in which these traditions are rooted.

2.4.1 Globalization and Hybrid Musical Awareness

The globalized nature of contemporary music education, as discussed in [17], introduces new paradigms for understanding the relationship between music, culture, and identity. While globalized education provides opportunities for hybrid musical awareness—where elements of different musical traditions blend seamlessly—it also underscores the importance of preserving the unique characteristics of individual traditions. *MravalJam* embodies this balance by presenting students with distinct cultural themes while allowing for creative exploration and experimentation.

For instance, in the space-themed funk level, students are exposed to the rhythmic complexity and melodic interplay characteristic of funk music. At the same time, they can experiment with layering sounds to create hybrid compositions that reflect their personal interpretations. This dynamic interaction mirrors the transcultural exchange discussed in [17], where cultural particularism and global hybridity coexist, enriching the learning process.

2.4.2 Broader Educational Implications

The inclusion of cultural representation in music education goes beyond teaching musical skills; it fosters critical thinking, empathy, and an appreciation for diversity. As a gamified platform, *MravalJam* bridges the gap between traditional and modern approaches to music education. By combining gamification mechanics with cultural enrichment, the project provides an engaging way for students to explore complex concepts such as harmony, rhythm, and melody within culturally diverse frameworks.

Moreover, the project's focus on global musical traditions addresses the concerns raised in the research [17] regarding the central tasks of music education in a multicultural

society. By introducing students to both the universality and particularism of musical traditions, *MravalJam* equips them with the tools to appreciate music as both a creative art form and a reflection of cultural identity. The insights from *"Re-appraising Ideas of Musicality in Intercultural Contexts of Music Education"* [16] and *"Music Education and Cultural Identity"* [17] reinforce the importance of integrating cultural representation into music education. By incorporating themes that span diverse musical traditions, *MravalJam* aligns with the call for an intercultural approach that transcends singular cultural perspectives. The project not only introduces students to foundational music theory but also fosters a deeper appreciation for the cultural and historical contexts of the music they engage with. This approach highlights the potential for music education tools to shape well-rounded, empathetic individuals who value diversity and creativity in an increasingly interconnected world.

2.5 Inspiration and Challenges in Music Education

The article *"Exploring the Global Decline of Music Education"* [1] highlights a disjuncture between the declining role of music education in schools and the cultural and economic significance of music in youth culture and the knowledge economy. The analysis identifies four key reasons for this decline: curriculum models that deprioritize music, an overemphasis on standardized evaluation, insufficient resources, and ineffective approaches to music advocacy. This global perspective resonates strongly with the challenges observed in our local educational context.

In many countries like Georgia, music education in most public schools has been relegated to what students and teachers alike perceive as "doing nothing" lessons. Music classes are often limited to dry historical overviews or uninspiring lectures, leaving students disengaged and detached from the subject. Practical exposure to music, the joy of creativity, and opportunities for self-expression are largely absent. As a result, music education fails to instill the cultural and artistic appreciation that is vital for shaping well-rounded, progressive citizens—an urgent need for a developing country like Georgia

with its rich, diverse cultural heritage.

Our personal experiences underscore these challenges. One member of our team has firsthand experience as a music educator, struggling with outdated curricula and unmotivated students. This frustration with traditional approaches, combined with the broader systemic issue of outdated and rigid school systems, inspired us to create *MravalJam*. Our project aims to address these problems by transforming music education into a fun, interactive, and gamified experience.

By using technology and game-based learning, *MravalJam* introduces students to fundamental music theory concepts in a way that feels approachable and engaging. The gamified environment motivates students through themed levels, incorporating global musical styles, while also reflecting Georgia's rich musical traditions. Integrating culture and art into gamified education fosters creativity, critical thinking, and a deeper appreciation for diversity—key traits for building a progressive society.

This effort is not just about improving music education but about leveraging it as a tool to ignite curiosity and engagement across subjects. As the article points out, addressing the challenges in music education requires innovative solutions that bridge the gap between music's significance in modern culture and its diminishing role in schools. *MravalJam* embodies this solution, combining modern educational technology with cultural enrichment to inspire the next generation of learners.

2.6 Summary

This chapter has explored the state-of-the-art in music education, highlighting prior works, digital tools, and educational game development methodologies that influenced the design of *MravalJam*. Key inspirations such as the *SeMI* project demonstrated the potential of interactive musical settings, while tools like *Yousician*, *Rhythm Cat*, *MuseScore*, and *Audacity* showcased how technology can make music education accessible and engaging. The study of game development platforms, including Unity and *Minecraft: Education Edition*, emphasized the effectiveness of gamification in fostering creativity and retention

among students.

Additionally, the incorporation of cultural representation into *MravalJam* was informed by research on intercultural perspectives and the relationship between educational system and cultural diversity. The project addresses global challenges in music education by providing an interactive, gamified, and culturally enriching platform.

By synthesizing these elements, *MravalJam* offers a new approach to music education that bridges traditional and modern methodologies, fosters cross-cultural appreciation, and addresses the declining role of music in educational systems worldwide. This foundation sets the stage for the subsequent chapters, which delve into the methodology, development, and implementation of the project.

Chapter 3

Methodology

3.1 Problem Description and Proposed Solution

Music education in many public schools, including those in Georgia, often lacks engagement and interactivity, leading to disinterest among students. Traditional approaches focus heavily on theoretical aspects or historical overviews without providing hands-on experiences, making it difficult for students to grasp these concepts easily, such as harmony, melody, and rhythm. Recognizing these limitations, *MravalJam* was developed to provide a gamified, interactive solution that makes music theory both accessible and enjoyable for primary to middle school students.

MravalJam aims to address these issues by leveraging game-based learning principles. The platform introduces students to foundational music theory concepts through interactive gameplay, where stickers representing sound layers are placed on a virtual board to create harmonious compositions. By focusing on engagement, replayability, and cultural enrichment, *MravalJam* provides an innovative approach to music education.

The project features two primary modes: levels mode and simulation mode. *Levels Mode* is a set of predefined challenges with specific stickers and goals tied to musical concepts. Each level offers a scoring system based on three core stats—harmony, melody, and rhythm. Additional objectives in some levels unlock achievement badges. *Simulation*

Mode is a freeform experience where all stickers are available, allowing students to experiment without predefined goals. This feature could be used for teachers to explicitly work on certain interactions between sound layers to better explain musical relations between them and concepts related to specific parts of music theory.

3.2 System Architecture and Design

MravalJam was developed entirely in Unity using C#. The system follows a modular architecture, with dedicated manager scripts handling specific functionalities. These managers interact independently and collaboratively based on gameplay needs, ensuring flexibility and maintainability.

The key components of the system include:

- **Stickers and Board**: Stickers represent sound layers, and placing them on the board plays corresponding audio tracks. Sound layers are in continuous loop, ensuring synchronization and simultaneously, listening to complex audio layers for extended amount of time helps with ear-training and musical recognition. The *BoardManager* communicates with the *SoundManager* to manage these interactions. *SoundManager* is responsible for creating and managing audio sources for stickers using Unity's built-in sound system. The sounds, recorded in Vorbis format, are synchronized by deciding the lengths of audio files appropriately to ensure seamless looping and alignment, while *GameManager* tracks player stats, determines star awards, and manages level progression. With stars, players get awarded with special badges in special levels. These levels have specific type of characteristic which is measured by a special-stat metric. This process of treating students with awards gives the sensation of achievement, motivating them to push forward. On top of positive reinforcement, this cements musical concepts in students' minds, helping them to remember and later on easily distinguish and appreciate compositions through specific characteristics.

- **Hint System**: Offers hints purchasable using HintKeys, which are earned by achieving stars. Purpose of hints in *MravalJam* is to lure player to right direction, while maintaining their attention to objectives and encouraging them not to give up. Hints will guide player along their musical journey, but will leave room for teacher interaction for more in-depth help.

- **Level Shuffling Mechanism**: Ensures replayability by shuffling the correct stickers available in each level. For example, from a pool of 10 correct stickers for a level, 5 are randomly selected while maintaining the whole structure of the music for each playthrough, leading to a better player experience. On top of that, the use of shuffling mechanisms adds variety to gameplay while maintaining thematic correctness.

The system is designed with modularity in mind. The use of singleton objects, such as *SoundManager* and *SaveManager*, ensures that only one instance of each manager exists throughout the game. For example, the *SaveManager* persists across scenes to manage player progress seamlessly. This allows us to have more centralized control. Our class diagram (Figure 3.2) illustrates the modular structure of the system, highlighting the relationships between the manager scripts and their interactions with gameplay elements like stickers and the board.

It is also worth mentioning, that all sounds were recorded by the development team, ensuring authenticity. Visual assets are a mix of generated and curated content.

3.3 Use Case Diagram and Class Diagram

The design of *MravalJam* is informed by both use case and class diagrams. These diagrams are essential tools in software development, providing a clear and structured representation of the system's functionality and architecture.

Use case diagrams are valuable in capturing the interactions between users and the system. They help identify user requirements, clarify system boundaries, and ensure that

all key functionalities are addressed. By focusing on user actions and system responses, use case diagrams simplify the communication of system requirements between developers and stakeholders.

Class diagrams, on the other hand, are a cornerstone of object-oriented design. They offer a detailed view of the system's structure by illustrating the relationships between classes, their attributes, and their methods. This visualization aids in designing a modular and scalable architecture, ensuring that components are reusable and maintainable. Class diagrams are particularly useful for identifying dependencies and interactions between system components, enabling efficient troubleshooting and updates.

Use Case Diagram

The use case diagram illustrates the interactions between users (students) and the system. Key use cases include: Level selection, placing stickers on the board, viewing and purchasing hints, unlocking stars and achievements and etc.

Figure 3.1 shows the use case diagram, highlighting the core functionalities of the system and the interactions between users and various components.

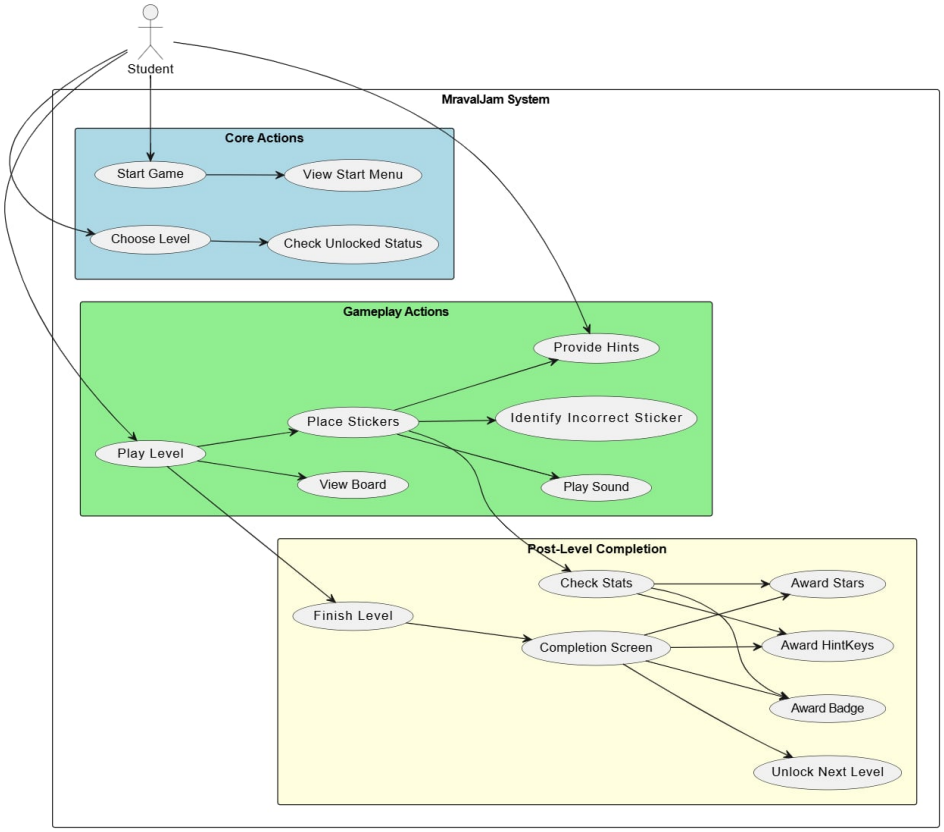

Figure 3.1: Use Case Diagram for *MravalJam*

Class Diagram

The class diagram highlights the modular structure of the system, detailing the relationships between core classes such as *SaveManager*, *UIManager*, *SoundManager*, and *GameManager*. It illustrates how these components operate independently while maintaining well-defined interactions.

Figure 3.2 depicts the class diagram, showcasing the modular architecture and emphasizing the system's design principles.

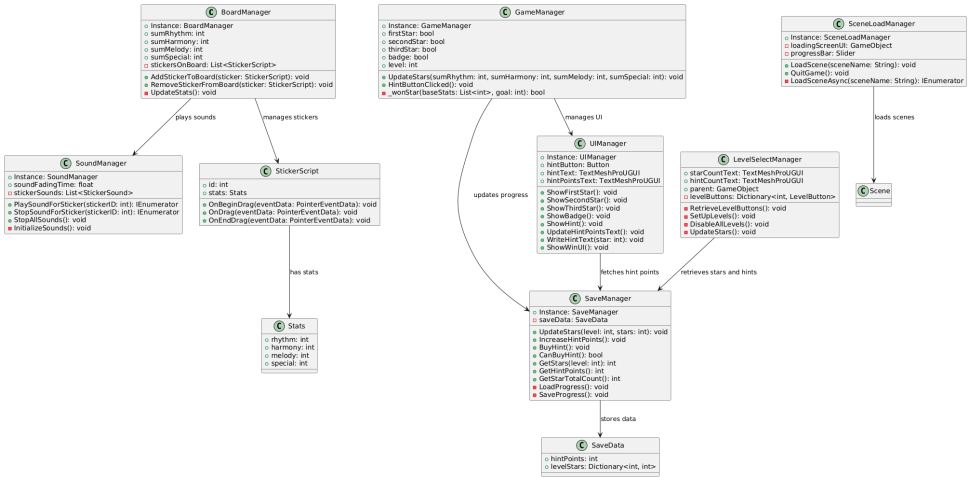

Figure 3.2: Class Diagram for *MravalJam*

3.4 Tasks and System Functions

The tasks implemented in *MravalJam* align with the educational and gamification goals. Each task is designed to contribute to the overall objective of making music education engaging, culturally enriching, and accessible to primary and middle school students. Below is a detailed explanation of the core tasks:

- **Engagement through Gameplay**: Students interact with stickers to explore and create harmonious compositions. By placing stickers on the virtual board, players unlock layers of music corresponding to harmony, melody, or rhythm. Each sticker is associated with a sound layer, and successful placement builds a rich musical composition. This task is designed to encourage hands-on exploration and creativity, keeping students engaged through immediate auditory feedback and dynamic soundscapes. The interaction with stickers also provides a playful learning experience, allowing students to experiment without fear of failure.

- **Educational Objectives**: Levels are meticulously designed to teach specific musical concepts, such as harmony, melody, rhythm, and occasionally additional cultural or stylistic elements. Progress is tracked through three primary stats—harmony,

melody, and rhythm—displayed in the User Interface (UI) for real-time feedback. Each level focuses on building the student's understanding of these elements through trial and experimentation. For instance, placing a wrong sticker may provide subtle auditory clues about dissonance, encouraging students to rethink and refine their decisions. The objective is to make learning intuitive while reinforcing theoretical knowledge through practical interaction.

- **Cultural Enrichment**: Themed levels introduce students to global musical traditions by incorporating culturally significant sounds and styles. For example, a Georgian-themed level immerses students in the region's unique melodic structures and rhythmic patterns, while a space-themed level introduces funky basslines and rhythms. These themes not only diversify the learning experience but also broaden students' cultural awareness, instilling an appreciation for global musical heritage. The cultural representation within levels provides a dual educational benefit, teaching students both music theory and cultural literacy.

- **Replayability**: Replayability is achieved through dynamic sticker shuffling and incremental level progression. In each level, stickers representing correct sound layers are randomly shuffled from a larger pool of options, ensuring that no two playthroughs feel identical. This mechanic encourages students to revisit levels for a fresh experience while solidifying their understanding of musical concepts. Additionally, progression through levels is tied to achieving stars based on performance in the harmony, melody, and rhythm stats. Students can replay levels to achieve higher scores, unlock achievements, or earn HintKeys, creating a loop that fosters continued engagement and learning.

3.5 Challenges, Difficulties and Their Solutions

Several challenges arose during the development of *MravalJam*, requiring innovative solutions. Firstly, sound synchronization problem arose. Ensuring that audio loops remained

aligned required precise control of sound file lengths and playback mechanisms in Unity. Besides that, balancing the difficulty and variety of replayable levels while maintaining educational objectives posed a challenge. Iterative testing and feedback helped refine the system. Lastly, crafting an interface suitable for primary to middle school students required balancing simplicity with functionality. Feedback from potential users guided improvements to ensure accessibility and engagement.

Implementing these solutions, the methodology behind *MravalJam* combines robust technical design with educational objectives. By leveraging modularity, gamification, and cultural enrichment, the project addresses critical gaps in traditional music education. Its architecture supports seamless functionality and replayability, while future expansions, such as new levels or multiplayer modes, can be easily integrated thanks to the modular design. Implementations will be further explored in Development chapter.

Chapter 4

Development

4.1 Overview

The development of *MravalJam* focused on creating an interactive, gamified platform that combines music education with cultural enrichment. The project was implemented entirely using Unity and C#, leveraging Unity's built-in functionality for all major features without the use of external libraries or packages. This decision ensured a streamlined development process, complete control over the system's architecture, and minimized potential compatibility issues.

This chapter provides an in-depth analysis of the key components, challenges faced, and solutions implemented during development. Emphasis is placed on the modular architecture, the use of Unity's features, and the iterative approach taken to refine the game's core mechanics, ensuring accessibility and engagement for primary to middle school students.

4.2 Key Components

4.2.1 Manager Scripts

The modular design of *MravalJam* revolves around manager scripts that handle specific aspects of the game. This separation of concerns allows for efficient debugging, scalability, and ease of maintenance. The following manager scripts form the backbone of the system:

SoundManager

The *SoundManager* in *MravalJam* is a core component responsible for managing all audio playback and synchronization through clearly defined functions. During the level initialization phase, the *InitializeAudioSources()* function is invoked to create and configure *AudioSource* components for all stickers, linking each sticker to a unique audio clip representing a sound layer. These audio layers are pre-synchronized during development to ensure seamless looping when played together. Initially, the *MuteAllSources()* function mutes all audio layers, and when a sticker is placed on the board, the *UnmuteSource(int stickerID)* function is called to unmute the corresponding audio source, creating a harmonious composition. Unity's *AudioSource* component is used for playback control, looping, and volume management. Additionally, the *ApplyFadeEffect()* function is implemented to handle fade-in and fade-out transitions, enhancing the overall auditory experience when activating or deactivating sound layers. Synchronization is achieved by invoking the *StartAllLayers()* function at the beginning of the level to start all audio sources simultaneously, even if muted. This method eliminates potential delays, ensuring consistent playback regardless of hardware performance. The SoundManager ties together the interactive elements of stickers and sound, delivering a cohesive and engaging educational experience.

BoardManager

The *BoardManager* in *MravalJam* is the central script that governs all interactions between stickers and the game board. Its primary role is to detect when a sticker is placed on the board and trigger the necessary updates across related game components. The *HandleStickerPlacement(int stickerID)* function is invoked whenever a sticker is placed on a valid board position. This function communicates with the *SoundManager* by calling *UnmuteSource(int stickerID)* to activate the corresponding sound layer, ensuring the audio aligns seamlessly with the gameplay. Additionally, the *UpdateStats()* function is executed to adjust key player statistics—harmony, melody, and rhythm—based on the properties of the sticker. By coordinating interactions between stickers, sounds, and statistics, the *BoardManager* serves as a foundational component that ensures a smooth and intuitive user experience while maintaining gameplay integrity.

GameManager

The *GameManager* script serves as the central controller for managing gameplay progression, star achievements, and hint functionality in *MravalJam*. It ensures seamless interaction with other components like the *UIManager* and *SaveManager*, enabling the game to respond dynamically to the player's actions. At its core, the *GameManager* tracks level-specific achievements, including the acquisition of three stars and a badge, which represent key gameplay objectives. These are stored as Boolean variables—*firstStar*, *secondStar*, *thirdStar*, and *badge*—and are updated dynamically through the *UpdateStars()* function. This function evaluates player progress based on three core stats: rhythm, harmony, and melody, along with a special stat for badge unlocking. It calls the private helper function *_wonStar()* to determine if a star goal is met, which requires all three base stats to meet or exceed a predefined threshold (e.g., 50%, 75%, or 100%). Upon achieving a star, the game triggers updates in the *SaveManager* to record the progress persistently and the *UIManager* to visually reflect these changes on the interface. For instance, stars are displayed using *ShowFirstStar()*, *ShowSecondStar()*, and *ShowThirdStar()*, while hint points

are updated using *UpdateHintPointsText()*. The *HintButtonClicked()* function handles hint system interactivity. It checks whether hints need to be purchased using accumulated hint points, facilitated by *SaveManager* methods like *CanBuyHint()* and *BuyHint()*. If a player lacks sufficient points, the hint button shakes using a coroutine, *ShakeHintButton()*, providing feedback to the user. Once hints are accessible, the *ShowHint()* function from the *UIManager* displays them to guide the player without directly solving the level. The script employs a singleton pattern, initialized in the *Awake()* method, to ensure a single instance of the *GameManager* is active at any time. This pattern centralizes game state management, allowing efficient communication with other components while preventing redundant instances. Overall, the *GameManager* orchestrates level progression, ensures accurate stat tracking, and enhances the user experience by coordinating hint functionality and visual feedback.

SceneLoadManager

The *SceneLoadManager* script is responsible for managing scene transitions and ensuring smooth loading experiences in the game. This is achieved using its singleton design pattern, which ensures only one instance of the manager exists across all scenes, maintained via the *Awake()* method where any duplicate instances are destroyed, and the *DontDestroyOnLoad()* function ensures persistence between scene transitions. The primary functionality revolves around its *LoadScene(string sceneName)* method, which initiates the asynchronous loading process by invoking the coroutine *LoadSceneAsync(string sceneName)*. The *LoadSceneAsync* method handles scene transitions by displaying the loading screen (via *loadingScreenUI*), monitoring the progress of the asynchronous operation using Unity's *SceneManager.LoadSceneAsync(sceneName)*, and updating a progress bar (if provided) for visual feedback. The script ensures user-friendly transitions by enforcing a minimum display time for the loading screen through *forcedLoadingTime* before hiding it. Additionally, it includes a *QuitGame()* method for cleanly exiting the application, further contributing to robust scene management. This modular approach ensures a seamless experience for players, particularly when navigating between levels or exiting the game.

SaveManager

The *SaveManager* script is responsible for managing the game's saving and loading functionalities, ensuring persistent storage of player progress across sessions. This script utilizes a singleton pattern for global accessibility and prevents duplication across scenes. During the *Awake()* method, the *SaveManager* initializes the save file path and attempts to load existing progress using the *LoadProgress()* method. If no save file exists, a new save data object is created. The script uses JavaScript Object Notation (JSON) serialization via the Newtonsoft.JSON library to write and read save data to and from a JSON file. Key functionalities include methods like *SaveProgress()*, which serializes the current save data and writes it to the file system, and *UpdateStars()*, which updates and saves the player's star progress, ensuring only the highest star count for a level is retained. The script also handles hint point management through methods such as *UpdateHintPoints()*, *IncreaseHintPoints()*, and *BuyHint()*, ensuring transactional consistency while enabling or preventing hint purchases based on available points. Getter methods like *GetStars()*, *GetHintPoints()*, and *GetStarTotalCount()* provide access to specific progress metrics for use in other game systems. With its robust design, the *SaveManager* ensures a seamless and reliable experience for saving and retrieving player data.

StatsManager

The *StatsManager* script is responsible for dynamically updating and displaying the player's progress in key statistics, specifically rhythm, harmony, melody, and optionally a special stat. It utilizes Unity's *TextMeshProUGUI* component to render text on the user interface. Upon initialization in the *Awake()* method, the script ensures that a single instance of *StatsManager* exists using the singleton pattern, destroying any duplicate instances. The primary method, *UpdateStats(int sumRhythm, int sumHarmony, int sumMelody, int sumSpecial)*, updates the text fields for each stat by converting the integer values into strings and assigning them to the corresponding *TextMeshProUGUI* objects.

Additionally, the method checks whether the optional *statSpecial* field is null before attempting to update it, ensuring compatibility across levels that may or may not include a special stat. This modular approach ensures a streamlined and consistent presentation of player progress throughout the game.

UIManager

The *UIManager* serves as the central hub for handling the user interface, ensuring clarity, accessibility, and responsiveness. At the core of its functionality, the *Awake()* method initializes the user interface by invoking *LevelGoalUISetup()* and *HintUISetup()*, which configure the level's visual goals and hint systems respectively. The hint system, powered by methods like *ShowHint()*, *HideHint()*, and *WriteHintText()*, allows players to access contextual hints tied to specific gameplay achievements while ensuring subtle guidance without providing outright solutions. Hints are dynamically displayed with smooth animations managed by the *FadeInPanel()* and *FadeOutAfterDelay()* coroutines, adding an interactive layer to the player's experience. For achievements such as stars or badges, the methods *ShowFirstStar()*, *ShowSecondStar()*, *ShowThirdStar()*, and *ShowBadge()* trigger animations and visual updates, utilizing *CanvasGroup* for transitions. The system also integrates player statistics updates via *UpdateHintPointsText()*, which syncs data from the *SaveManager*. Furthermore, usability-focused features like the *ShakeHintButton()* coroutine add vibrational feedback to elements like the hint button, ensuring player engagement. The UI dynamically adjusts its display to reflect real-time progress, victory conditions, and player interactions through comprehensive methods ensuring a seamless and immersive user experience.

4.2.2 Sticker and Sound Integration

Stickers are the core interactive elements of *MravalJam*, representing sound layers that players use to create harmonious compositions. Each sticker is linked to an audio file via the *SoundManager*, and their interaction with the board triggers corresponding sound

layers. Correct stickers contribute to the intended composition of each level, while incorrect stickers produce dissonance. This design encourages experimentation, helping players develop an intuitive understanding of sound layering and harmony.

To ensure seamless audio playback all audio files were preprocessed to have matching lengths, ensuring they loop without breaks or inconsistencies. Unity's *AudioSource* components were used to manage playback, with all sound layers initialized simultaneously but muted until activated by sticker placement. Audio synchronization was achieved by starting all tracks on the same frame and dynamically unmuting them as stickers are placed. This approach eliminates latency issues and maintains the integrity of the compositions, even on lower-end devices.

Replayability was a key consideration during development. Each level contains a pool of around 10 correct stickers, with roughly 5 randomly selected and shuffled every time the level is loaded. This ensures that each playthrough feels fresh while maintaining thematic consistency.

The sticker shuffling mechanism involves: Predefined categorization of stickers into pools of "correct" and "incorrect" options; Randomized selection of stickers while ensuring musical compatibility between chosen combinations; Dynamic assignment of audio sources to the selected stickers at runtime, ensuring seamless integration with the *SoundManager*.

4.3 Challenges and Solutions

One of the most significant challenges was ensuring that multiple audio layers remained synchronized, especially during loop transitions. To solve this issue, Audio files were carefully edited to ensure uniform lengths and consistent tempo. All audio tracks were initialized at the same time but muted until activated, reducing latency and synchronization issues. Unity's *AudioSource.loop* property was leveraged to maintain seamless playback.

Secondly, designing a shuffling mechanism that maintained thematic and musical integrity required extensive testing. Predefined pools of stickers were carefully curated to

ensure that any combination of selected stickers resulted in a harmonious composition. This process involved multiple iterations of testing and refinement.

Lastly, crafting an intuitive interface for young students required balancing simplicity with functionality. Several iterations were conducted to refine the layout, color scheme, and navigation elements. Key improvements included: Designing menu structures to make it easier for users to navigate and make decisions without feeling overwhelmed; On top of that, it is important to make sure to add simple tutorial to teach students how to properly play the game.

4.4 Technical Solutions

Unity's native features were utilized extensively to ensure efficiency and stability. Key functionalities include: *AudioSource* component for managing sound playback and synchronization; *Canvas* system for creating and updating user interfaces dynamically; Unity's scene management tools for transitioning between levels with loading screens; The *Prefab* system for efficiently managing reusable assets like stickers and UI elements.

All audio tracks were recorded by the development team to ensure authenticity and alignment with educational objectives. These tracks were processed to maintain consistent quality and seamless looping. Visual assets were created or curated to match the thematic elements of each level, ensuring that the visuals complemented the auditory experience.

No external open-source code or libraries were used in the development of *Mraval-Jam*. All scripts and assets were created in-house or generated using Unity's default tools, ensuring complete control over the development process and avoiding unnecessary dependencies.

4.5 Summary

The development of *MravalJam* focused on creating a seamless and engaging learning experience through modular architecture, custom assets, and innovative gameplay mechanics. Challenges such as sound synchronization, replayability, and UI design were addressed through iterative testing and creative problem-solving. By leveraging Unity's built-in features, the project achieved its educational and gamified objectives without relying on external libraries. The result is a polished and scalable platform that lays the foundation for future enhancements and broader accessibility.

Chapter 5

Performance Evaluation

This chapter outlines the tests conducted to evaluate whether *MravalJam* fulfills its objectives and addresses the challenges identified in the Analysis and Methodology chapter. Each test is described with its purpose, methodology, and expected results, followed by an analysis of what was learned, what could have been done differently, and how the results align with or exceed the initial objectives.

5.1 User Interface Testing

The goal of UI testing was to ensure that *MravalJam* provides an intuitive and engaging interface for primary to middle school students. The focus was on clarity, color themes, and overall navigational simplicity to accommodate young learners with minimal technological experience.

UI prototypes were iteratively tested with peers. Testing involved: Evaluating button placement and sizing to ensure ease of interaction; Testing color schemes for clarity, avoiding overwhelming or overly vibrant designs; Simplifying menu navigation to make levels, simulation mode, and settings easily accessible.

The final design adopted a minimalistic and user-friendly approach, incorporating vibrant yet balanced colors to maintain engagement. Feedback indicated that the color schemes and layouts effectively guided users through the gameplay without confusion.

Simple UI designs resonate well with younger audiences, and iterative testing is crucial to identify potential accessibility issues. A better approach would have been more extensive testing with actual primary and middle school students, which could provide additional insights into real-world usability challenges. The process of refining UI layouts and color schemes add an additional layer of polish to the project, ensuring a higher standard of accessibility.

5.2 Sound Synchronization Testing

The primary objective was to achieve seamless synchronization of audio layers to ensure that sound compositions remain cohesive and musically accurate, regardless of the sticker arrangement or order of interaction.

Various synchronization approaches were tested within Unity's default AudioSource framework. At first, activating each layer one by one was attampted, introducing timing mismatches for low-end users. Sounds were synchronized based on calculated time-passed values for each bar, but this added computational complexity and was vulnerable to timing issues. Then, different approach was used, which made sure that all layers were synchronized from the start, muted initially, and unmuted dynamically during gameplay.

The preloaded muted layers approach yielded the most reliable and musically accurate results, ensuring that layers always aligned perfectly regardless of user input timing. Starting all layers simultaneously and muting/unmuting dynamically is an effective synchronization method for educational and gamified applications. Future iterations might explore advanced audio middleware, such as FMOD [18] or Wwise [19], for even more precise and dynamic sound control.

5.3 Musical Complexity Testing

The goal was to determine the appropriate level of musical complexity for the target audience while maintaining cultural and educational value.

Musical themes were tested iteratively to balance simplicity and authenticity. Experiments were conducted with simplified melodies, rhythms, and harmonies for easier digestibility. Also, whether overly simplified compositions compromised cultural or musical integrity was determined. Lastly, levels were tested with peers to gauge the overall appeal and accessibility of the music. The selected musical themes struck a balance between accessibility and richness, ensuring that students could grasp the concepts while appreciating the diversity and depth of the compositions.

Maintaining cultural and musical significance while simplifying elements for educational purposes is achievable with careful iteration and peer feedback. Although, direct feedback from children within the target demographic would enhance the alignment of musical complexity with their preferences and learning capacities.

5.4 Hint and Reward System Testing

The goal was to develop a hint and reward system that motivates students without making the gameplay trivial or overly challenging.

Several iterations of hints and reward systems were tested. Development team set out to create hints that provided subtle guidance rather than explicit answers. With that, it was taken into consideration that adjusting the difficulty of obtaining HintKeys and achieving stars to maintain balance was quite important too. Decision was made to perceive value of badges and achievements as motivational tools, just so it would keep players engaged. Again, maintaining balance between difficulty of the challenge and reward for completion was crucial. Of course, our results can be improved by testing application on target sector - primary to middle school students.

In conclusion, an effective hint system should complement learning objectives without reducing the educational value of the gameplay. Rewards tied to performance foster engagement and encourage repeated play. Furthermore, testing the hint system with music or art teachers could refine its ability to align with teaching goals and identify areas for improvement.

5.5 Discussion and Future Directions

MravalJam successfully addressed several core objectives. It delivered an intuitive and engaging UI design. It also achieved reliable sound synchronization for seamless musical compositions. Maintaining cultural and educational value in musical themes was managed. Last but not least, designed an effective hint and reward system to balance challenge and motivation.

Key lessons learned included the importance of iterative testing, while underlining the need to involve the target demographic in future testing phases. While not initially planned, the focus on refining the hint and reward system, and also coming up with a solution for sound synchronization, added significant value to the project, improving its overall quality and user experience. But, of course, there still is room for improvement. For example, as mentioned before, the lack of direct testing with primary and middle school students due to logistical constraints limited the ability to gather authentic user feedback. This highlights the need for future iterations to prioritize real-world testing and more importantly - teacher involvement.

From testing and then discussing results couple of key objectives to be met in future development were derived. Firstly, expanding the pool of testers to include the target demographic. Secondly, integrating advanced audio middleware for enhanced sound experience and adding new creators for more diverse level design and music composition. Lastly, exploring additional reward mechanisms to further gamify the experience.

Chapter 6

Conclusions

The development of *MravalJam* represents a significant step toward transforming music education for primary to middle school students by combining gamification, interactivity, and cultural enrichment. This project was motivated by the pressing need to address the disengagement and lack of interactivity in traditional music education methods, particularly in public school systems like those in Georgia. Through its innovative approach, *MravalJam* offers students an accessible and enjoyable platform to explore foundational music theory concepts such as harmony, melody, and rhythm while fostering an appreciation for diverse musical traditions.

Inspired by prior works such as *Yousician*, *Rhythm Cat*, and *Minecraft: Education Edition*, *MravalJam* incorporates key elements of gamified learning while addressing their limitations. Unlike these tools, which focus on individual skill-building or general creativity, *MravalJam* emphasizes thematic learning through interactive sticker-based gameplay. Additionally, the system draws from cultural integration perspectives by introducing levels themed around global musical styles. These influences informed the project's unique design, bridging gaps in both music education and cultural representation.

The implementation of *MravalJam* leveraged Unity's built-in functionality, resulting in a modular and scalable architecture. Key technical solutions, such as synchronized sound layers, dynamic sticker shuffling, and a user-friendly interface, were developed to enhance the player experience and ensure replayability. Challenges encountered during the

development, including sound synchronization and balancing replayability, were effectively addressed through innovative problem-solving.

Looking forward, the potential for expanding *MravalJam* is substantial. Future work could include adding more themed levels to represent a wider variety of musical traditions and instruments, enhancing the hint system to provide adaptive learning pathways, or introducing multiplayer features to encourage collaborative learning. Furthermore, expanding platform compatibility to include mobile devices could broaden accessibility and impact.

In conclusion, *MravalJam* is more than just an educational tool—it is a testament to the power of technology and gamification in reshaping traditional educational models. By engaging students in interactive and culturally enriching activities, the project not only enhances their understanding of music but also fosters creativity, curiosity, and empathy. This work lays the groundwork for future innovations in educational technology, with the potential to inspire similar projects across other disciplines.

Bibliography

[1] J. L. Aróstegui, "Exploring the global decline of music education," *Arts Education Policy Review*, vol. 117, no. 2, pp. 96–103, 2016. DOI: 10.1080/10632913.2015.1007406. eprint: https://doi.org/10.1080/10632913.2015.1007406. [Online]. Available: https://doi.org/10.1080/10632913.2015.1007406.

[2] D. J. Teachout, "The impact of music education on a child's growth and development," *Sounds of learning. Carlsbad, CA: International Foundation for Music Research*, 2005.

[3] https://unity.com/, Unity is a cross-platform game engine developed by Unity Technologies, first announced and released in June 2005 at Apple Worldwide Developers Conference as a Mac OS X game engine. The engine has since been gradually extended to support a variety of desktop, mobile, console, augmented reality, and virtual reality platforms.

[4] M. Cardoso and R. P. Lopes, "Interactive musical setting with deep learning and object recognition," in *12th International Conference on Computer Supported Education (CSEDU 2020)*, SCITEPRESS, vol. 1, 2020, pp. 663–667.

[5] https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Redmon_You_Only_Look_CVPR_2016_paper.pdf, You Only Look Once (YOLO) is a series of real-time object detection systems based on convolutional neural networks.

[6] https://yousician.com/, Yousician makes learning to play an instrument fun and easy, no matter your skill level.

[7] https://melodycats.com/rhythm-cat/, Rhythm Cat is an entertaining game with a great soundtrack to play along to, but it also has the added benefit of teaching you how to read the most commonly used music rhythms, with levels that build up progressively.

[8] https://musescore.org/en, notation editor with easy input via mouse, keyboard or MIDI system.

[9] https://www.audacityteam.org/, Audacity is the world's most popular free software for recording and editing audio. So if you're producing music, a podcast, or just playing around with audio, Audacity is for you.

[10] https://garagebandonpc.com/, GarageBand is a software application by Apple for macOS, iPadOS, and iOS devices that allows users to create music or podcasts. It enables users to create multiple tracks with software synthesizer presets (to be played on a MIDI keyboard and/or sequenced on a piano roll), pre-made and user-created loops, an array of various effects, and voice recordings.

[11] https://www.soundtrap.com/musicmakers, Soundtrap is a freemium online cross-platform digital audio workstation (DAW) for browsers that allows users to create music or podcasts.

[12] E. Parkita, "Digital tools of universal music education," *Central European Journal of Educational Research*, vol. 3, no. 1, pp. 60–66, Apr. 2021. DOI: 10.37441/cejer/2021/3/1/9352. [Online]. Available: https://ojs.lib.unideb.hu/CEJER/article/view/9352.

[13] O. Comber, R. Motschnig, H. Mayer, and D. Haselberger, "Engaging students in computer science education through game development with unity," in *2019 IEEE Global Engineering Education Conference (EDUCON)*, 2019, pp. 199–205. DOI: 10.1109/EDUCON.2019.8725135.

[14] *Minecraft: Education edition*, https://education.minecraft.net/en-us/discover/impact, Minecraft Education prepares students for the future, building future-ready skills like creativity, problem solving, and systems thinking, and nurturing a passion for play.

[15] *Minecraft*, https://www.minecraft.net/en-us, Minecraft is a 2011 sandbox game developed and published by Swedish video game developer Mojang Studios.

[16] J. O'flynn, "Re-appraising ideas of musicality in intercultural contexts of music education," *International Journal of Music Education*, vol. 23, no. 3, pp. 191–203, 2005. DOI: 10.1177/0255761405058238. eprint: https://doi.org/10.1177/0255761405058238. [Online]. Available: https://doi.org/10.1177/0255761405058238.

[17] R. A. Davis, "Music education and cultural identity," *Educational Philosophy and Theory*, vol. 37, no. 1, pp. 47–63, 2005. DOI: 10.1111/j.1469-5812.2005.00097.x. eprint: https://doi.org/10.1111/j.1469-5812.2005.00097.x. [Online]. Available: https://doi.org/10.1111/j.1469-5812.2005.00097.x.

[18] https://www.fmod.com/, FMOD is a proprietary sound effects engine and authoring tool for video games and applications developed by Firelight Technologies. It is able to play and mix sounds of diverse formats on many operating systems.

[19] https://www.audiokinetic.com/en/wwise/overview/, Wwise (Wave Works Interactive Sound Engine) is Audiokinetic's software for interactive media and video games, available for free to non-commercial users and under license for commercial video game developers. It features an audio authoring tool and a cross-platform sound engine.

Appendix A

Project proposal

INSTITUTO POLITÉCNICO DE BRAGANÇA
Escola Superior de Tecnologia e Gestão

Curso de Licenciatura em Engenharia Informática
3rd Year Project - School year of 2023/2024

MravalJam - Interactive Musical Setting Vol.2

	Artificial intelligence		Multimedia/Augmented reality/…
	Networks and systems management		Mobile apps
	Web platforms	√	Desktop apps

* - Other area

Supervisor: Rui Pedro Lopes

Co-supervisor: Ahmed Gamal Ibrahim

1 Research problem

Enhancing Music Education Through Gamified Experience.

2 Solution proposal

Create application with capabilities of using stickers/board to create imagery and play appropriate sound textures.

3 Project development strategy

The MravalJam project explores the potential of gamified music education to address the challenges of engaging school students in learning more about music. The platform is built using the Unity Game Development Environment and introduces students to foundational musical concepts through an interactive sticker-based system.

Team size:	2 students (Davit Chikobava, Giorgi Maglakelidze)
Required resources:	