MLopoly: a Machine Learning "Mod"

Seyed Mahdi Seyedishandiz[1][0009-0002-7144-5771] (✉), Amirmohammad Dalvand[1][0009-0007-5251-7342], Sina Gholami Fashkhami[1][0009-0003-2122-7896], Amirali Hoseinnatajaghamaleki[1][0009-0006-3516-826X], Chiara Eva Catalano[2][0000-0003-1997-9143] and Francesco Bellotti[1][0000-0003-4109-4675]

[1] Department of Electrical, Electronic and Telecommunication Engineering (DITEN), University of Genova

[2] Istituto di Matematica Applicata e Tecnologie Informatiche (CNR IMATI) - Genova

mahdiseyyedi98@gmail.com

Abstract. As machine learning (ML) instruction is dramatically widened its audience, new tools and methods are needed to allow an effective comprehension also by students without a strong mathematic/programming background, who may have more difficulty in districting among the fundamental theoretical concepts. To address this gap, we propose MLopoly, a Monopoly-inspired serious game designed to support interactive and experiential learning of core ML topics. The game maps decision-making challenges across realistic scenarios (e.g., restaurant, farm, bank) where players iteratively build, tune, and evaluate ML models. Game mechanics such as property upgrades, rent calculations, and challenge cards are directly tied to players' performance on ML tasks. We discuss game design, gameplay dynamics and implementation issues, A preliminary validation study has been carried out conducting a 20-item questionnaire based on Bloom's Taxonomy and Attention, Relevance, Confidence, and Satisfaction (ARCS) Motivation Model to assess learning and motivation. Results indicate that MLopoly offers an engaging, low-barrier environment for understanding and applying ML, also encouraging disciplinary deepening of the covered topics.

Keywords: Serious game design, Machine Learning teaching Decision Trees, Monopoly, Hyperparameter Tuning

1 Introduction

As Machine Learning (ML) has become a highly-valued and attractive academic discipline, state of the art gaming technologies offer an opportunity to motivate, introduce and support students, complementing instructional tools and methods [1, 2]. An abstract and technical approach to fundamental concepts (e.g., classification, model training, overfitting, and hyperparameter tuning) may be overwhelming for beginners and discourage engagement, forming a barrier to the discipline.

By embedding gameplay elements such as challenges, feedback, choice, and immersion into the learning process, serious games create an experiential and motivating context for grasping complex ideas [3]. This approach is valuable for STEM disciplines, as learners benefit from safe, iterative experimentation and the ability to visualize and manipulate key concepts in action.

We introduce MLopoly, a Monopoly-inspired serious board game designed to teach fundamental ML concepts, the first release focuses on the Decision Tree algorithm -

through interactive gameplay. MLopoly aims to become a pleasant tool, useful for students by allowing them to directly engage with fundamental, abstract concepts like model complexity, data splitting, overfitting, and hyperparameter tuning in a playful and intuitive environment. To advance in the game, players are called to manage a thorough machine learning workflow process, progressing through data collection, training, validation, and testing phases, tuning their models and observing performance across various application scenarios.

MLopoly uses tangible metaphors to make abstract ML concepts concretely applicable and meaningful. For example, players acquire and upgrade model "properties", use Decision Cards to adjust model parameters, and earn performance scores as feedback. In this way, the game leverages the motivational and cognitive benefits of game-based learning to improve understanding and retention of ML concepts. In the following, we present the design rationale, implementation, and evaluation of MLopoly. Submitting a questionnaire based on Bloom's Taxonomy and Attention, Relevance, Confidence, and Satisfaction (ARCS) Motivation Model, we assessed the game effectiveness in helping players internalize key ML principles through structured gameplay and compelling decision-making.

2 Related Work

Typical ML education - often lecture- or code-based – may be too abstract and difficult, particularly for students without strong programming or math backgrounds [1]. To address these limitations, researchers have increasingly explored serious games as engaging alternatives for teaching AI and ML concepts.

One example is the Learn to Machine Learn project by Voulgari et al. [1], which developed ArtBot [4], a narrative-based game that introduces supervised and reinforcement learning to younger students. ArtBot emphasizes algorithmic transparency and student engagement through interactive, visually guided feedback and simplified decision trees.

Other games, like ViPER [5], also aim to foster AI literacy but tend to abstract away algorithmic detail, offering limited direct interaction with parameters or evaluation metrics [1]. As a result, key applied concepts such as hyperparameter tuning, model evaluation (e.g., F1-score, RMSE), or regression/classification strategies remain underexplored in current ML game-based curricula.

Beyond digital platforms, Monopoly-style board games have been adapted to educational settings, such as psychology and finance, to facilitate learning through structured play [6, 7]. Among these, REV-OPOLY [8] stands out for incorporating Augmented Reality and multiplayer elements to enhance engagement and knowledge retention. Students reported high satisfaction and improved outcomes compared to traditional revision methods [7].

Building on these foundations, MLopoly is introduced as a serious board game designed to teach core ML concepts, particularly decision trees, through scenario-based challenges (e.g., restaurant, bank, farm). Unlike prior work, MLopoly integrates active hyperparameter control (e.g., max_depth, min_samples_split), iterative model testing, and domain-specific feedback mechanisms. It distinguishes itself from ArtBot by enabling deeper model tuning, and from REV-OPOLY by embedding evaluation metrics and diagnostic feedback into its gameplay loop. In doing so, MLopoly addresses a critical gap in ML education: providing students not just with conceptual familiarity, but hands-on, structured experience in model development and tuning within an accessible game format.

3 Game design & Methodology

3.1 Game Structure and Metaphor

Each side of the board is explicitly linked to a specific business domain (restaurants, farms, supermarkets, banks) (See Fig. 1), and each domain is directly associated with both a machine learning task and the tuning of a particular hyperparameter. For example, the restaurant side involves a regression task where the key focus is adjusting the max_depth parameter of a decision tree.

This mapping was designed to provide players with concrete challenges while simultaneously linking gameplay to the effect of hyperparameters on different ML tasks. In this way, each business context serves not just as a thematic setting, but as an accessible metaphor to illustrate how changes in model configuration impact performance outcomes. All sides with their machine learning task and particular hyperparameter are listed below:

- Regression with max_depth (restaurants)
- Classification with max_depth (farms)
- Regression with min_samples_split (supermarkets)
- Classification with min_samples_split (banks)

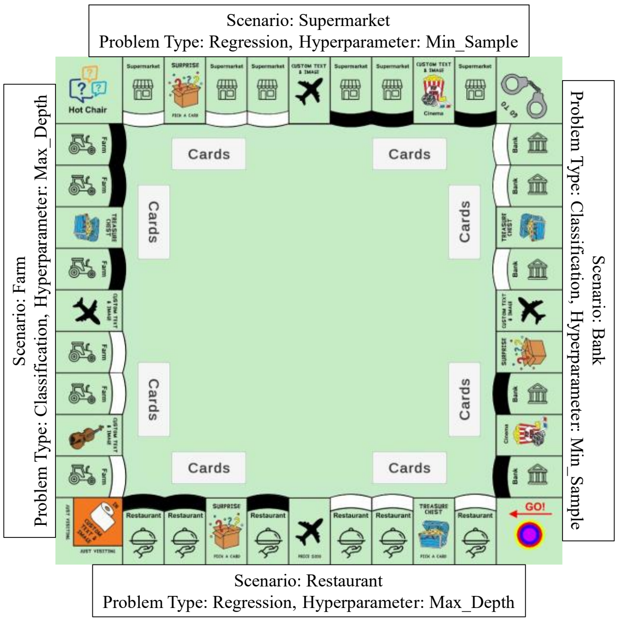

Fig.1. The MLopoly board

The ownership and upgrade systems are directly tied to how well the player understands and applies machine learning concepts. Performance is not based on luck or capital alone, but on measurable understanding and practical modeling decisions.

MLopoly simulates a machine learning workflow, embedding regression and classification tasks into the different domain-themed tiles. When landing on a tile, players are presented with a structured ML problem relevant to the domain, either a regression (e.g., predicting customer return frequency based on features) or classification task (e.g., predicting crop success based on soil and rainfall conditions).

Each challenge prompts players to build and tune a Decision Tree model (See Fig. 2), initially with default hyperparameters (e.g., max_depth = 2). Based on their decisions, the game provides immediate feedback using performance metrics relevant to the task type, such as RMSE for regression or F1-score/confusion matrix for classification. (Fig. 3)

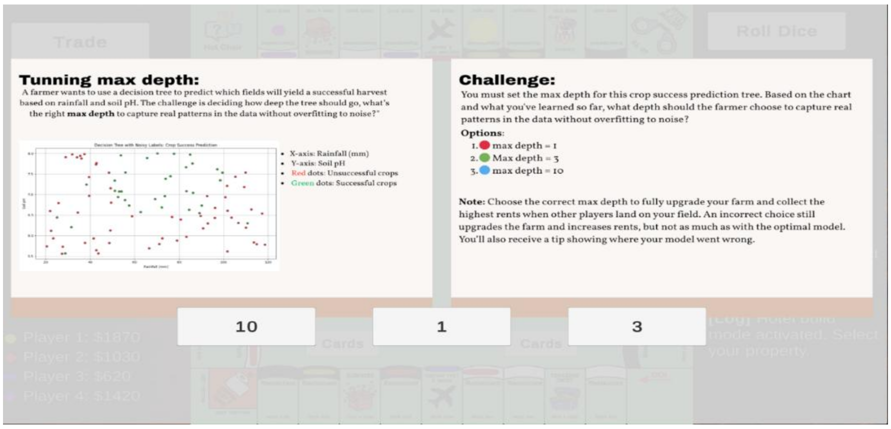

Fig.2. Tuning hyperparameters through Q&A

Fig.3. Feedback on the player's choices

This feedback guides players to iterate on their model design by adjusting hyperparameters like max_depth or min_samples_split.

Performance improvements (e.g., a reduced RMSE or improved recall) unlock in-game rewards, including model upgrades, property enhancements, or rent multiplier, mechanically linking model quality with economic advantage in gameplay. In parallel, short conceptual challenges (e.g., multiple-choice questions on evaluation metrics) allow players to consolidate understanding and earn strategic bonuses.

The game structure encourages experimental learning, where players iteratively refine models and observe the trade-offs involved in ML design decisions, such as overfitting due to excessive depth or misclassification caused by shallow models. These mechanics make abstract concepts like generalization and metric-based optimization more tangible and retainable through repeated play.

3.2 Learning approach and content

The core educational content centers on the Decision Tree algorithm, selected for its interpretability and pedagogical accessibility. Within the game, players engage with the structure and logic of decision trees, including the role of nodes, splits, and leaf predictions—as they iteratively build models on regression and classification problems.

A major instructional focus is placed on hyperparameter tuning, especially max_depth and min_samples_split, as these directly influence the trade-off between model complexity and generalization. Players are required to make tuning decisions based on the behavior of their model on simulated validation data, thereby experiencing firsthand the bias-variance trade-off in action. For example, increasing max_depth may improve performance on training data, but risks overfitting, a tension visualized through changing game outcomes such as fluctuating rent values.

Model evaluation metrics are embedded contextually: players working with regression tasks (e.g., restaurant or supermarket scenarios) are introduced to RMSE, MSE, and R^2, while those addressing classification problems (e.g., farms or banks) engage with confusion matrices, precision, recall, and F1-score. Each domain emphasizes metrics aligned with realistic concerns, for example, high recall in crop prediction (to avoid missing a productive yield) or precision in banking (to avoid false loan approvals).

The overarching pedagogical objectives of MLopoly include:

- Enabling learners to understand how decision trees generate predictions.
- Differentiating between regression and classification tasks via gameplay.
- Experimenting with hyperparameter values and observing their impact.
- Applying evaluation metrics appropriately to scenario-driven challenges.
- Reinforcing understanding through feedback loops and reflective tuning.

The pedagogical design of MLopoly is rooted in constructivist and experiential learning theory [6], which emphasizes learning through doing, reflection, and contextual engagement. This foundation supports a shift from passive absorption of ML theory to active, iterative exploration of models and metrics. One core design element is constructive alignment: the game mechanics are purposefully linked to intended learning outcomes. For instance, when players tune max_depth and observe changes in rent (reflecting model evaluation), they are actively engaging with the concept of overfitting. This alignment ensures that every game action contributes meaningfully to concept acquisition.

To facilitate scaffolded learning, the game deploys a tiered system of "Educational Cards" that gradually increases complexity. Each domain features a series of three cards that introduce core ideas sequentially beginning with model structure, followed by tuning strategies, and culminating in metric-based evaluation. This design supports progressive internalization of ML concepts and allows players to revisit and reinforce earlier material.

Feedback is contextualized through consequences in the game economy and narrative. A poorly tuned model may lead to low rental income or missed opportunities, while strong model performance (quantified via evaluation metrics) unlocks property upgrades and rewards. These mechanics simulate the stakes of real-world ML decisions and embed learning within meaningful game outcomes.

Furthermore, the game encourages learners to compare model performance over time - across different sectors and tuning strategies - promoting reflection on generalization and adaptation. For example, players may notice that deeper trees improve performance in some domains but fail in others, implicitly reinforcing the notion that model effectiveness is context dependent.

3.3 Implementation

We built MLopoly using the Unity game engine (version 2022.3.6f1 LTS) with C# scripting, leveraging Unity built-in UI, animation, and scene management system [9]. Its overall design mirrors a classic Monopoly board: each colored property set corresponds to a specific ML scenario and focuses on tuning a particular decision tree hyperparameter. This familiar board layout provides a structured framework where acquiring properties equates to engaging with different ML tasks.

Gameplay progresses as players move around the board, acquire properties, and tackle ML tasks in each domain. Owning all properties in a color group unlocks a mini ML challenge for that domain. These challenges are dataset-driven problems or multiple-choice quizzes presented through Unity UI system. To simulate model evaluation, the game provides visual feedback with decision tree diagrams and evaluation metrics . Each domain uses a unique dataset, ensuring varied data distributions and problem settings for realism. An interactive trading system is included to preserve strategic decision-making akin to classic Monopoly. Players can trade properties via an in-game menu to negotiate domain coverage. Trades execute automatically to maintain smooth game flow and prevent interruptions to the learning experience.

MLopoly emphasizes clear, consistent visual feedback to reinforce learning. Unity 2D graphics renders the Decision Tree structure and decision boundaries, and illustrates concepts like overfitting, underfitting, or well-tuned models. Gameplay interactions are event-driven using Unity event system, enabling natural actions such as dice rolls, player movement, and property management. During key decisions, context-sensitive pop-up messages explain the underlying ML mechanisms, helping players connect their in-game actions to ML outcomes. An internal simulation module computes the impact of player-selected hyperparameters on synthetic datasets. It generates mock evaluation results (performance scores, model behavior outcomes), which are dynamically shown as visual overlays and directly influence in-game consequences (e.g., adjusting in-game rewards or penalties based on model performance). One particularly engaging feature is the “Hot Chair Challenge,” a timed quiz competition between two players. This head-to-head quiz tests recall and decision speed under time pressure, reinforcing key concepts in a funny, competitive way. All quizzes and outcomes are handled through a modular system, making it easy to expand content with new questions or topics.

During the game, ML concepts are introduced gradually through gameplay rather than overwhelming players with mathematical formalism upfront. Each new concept is first presented via a teaching card or scenario-specific example, allowing players to learn incrementally. Hyperparameter tuning, for example, is transformed from an abstract idea into a tangible game mechanic with clear consequences: players see how tuning affects their "model" and game progress, reinforcing its importance through immediate feedback and iterative practice.

4 Experimental results

To evaluate MLopoly effectiveness we conducted a mixed-methods user study combining in-game behavioral metrics with a post-game questionnaire [3]. The study was grounded in two widely recognized educational frameworks targeting mid-level cognitive skills: Bloom's Revised Taxonomy [10, 11] and the ARCS Motivation Model [12]. We designed a 21-item Likert-scale questionnaire, aimed at assessing perceived learning (Part A) and motivational engagement (Part B). Additionally, three open-ended questions allowed participants to share reflections on the game design, usability, and educational impact. The full list of questions is shown in Table 1. Also, all the questions and results are accessible through [13].

The study involved university students, with various academic backgrounds. In total, 24 participants took part in the study (10 females, 14 males), with an average age of 27 years. While all had basic math knowledge, their prior exposure to ML varied, allowing us to observe how MLopoly performs across different levels of experience.

Before each session, participants received a short introduction to ML concepts (e.g., decision trees, terminology of training/validation/test data) to ensure a smoother start with the game.

Results (Fig. 4) show strong alignment between the game objectives and self-reported learning. For instance, 83% of participants agreed or strongly agreed that they could recall key ML concepts about Decision Trees (Q1). Over 75% of the respondents reported better understanding of hyperparameter distinctions (Q2) and data partitioning (Q7). Around 70% stated they could now explain the effect of decision tree hyperparameters like max_depth and min_samples_split (Q8) and identify issues like overfitting and underfitting (Q4). These outcomes were supported by qualitative comments such as: "I now understand how depth affects accuracy," and "The decisions I made in the game helped me see how models learn."

Table 1. Questionnaire

ID	Text
Q 1	I can recall key concepts about decision trees after playing the game.
Q 2	The game helped me understand the difference between parameters and hyperparameters.
Q 3	I was able to apply the knowledge from the game to real-world examples.
Q 4	I can identify whether a model is underfitting, overfitting, or well-fitted.
Q 5	I feel confident evaluating the performance of a machine learning model.

Q 6	I feel able to build a basic decision tree with the knowledge I gained.
Q 7	The game helped me recognize the purpose of training, validation, and test data.
Q 8	I can explain the effect of hyperparameters like max_depth and min_samples_split.
Q 9	I now understand how machine learning can be used in different domains.
Q 10	I was able to reason about how the model made decisions during the game.
Q 11	The game grabbed my attention and kept me focused.
Q 12	The game content was interesting and enjoyable.
Q 13	I found the game's content relevant to my learning goals.
Q 14	I saw a clear connection between game mechanics and machine learning concepts.
Q 15	I felt capable of completing the game tasks.
Q 16	The difficulty level of the game was appropriate for my skills.
Q 17	I was satisfied with what I learned through the game.
Q 18	I felt a sense of accomplishment after finishing the game.
Q 19	I would be interested in learning other topics through similar games.
Q 20	This game encouraged me to explore more about machine learning after the session.
Q 21	I was able to make meaningful choices in the game that affected my learning outcome.

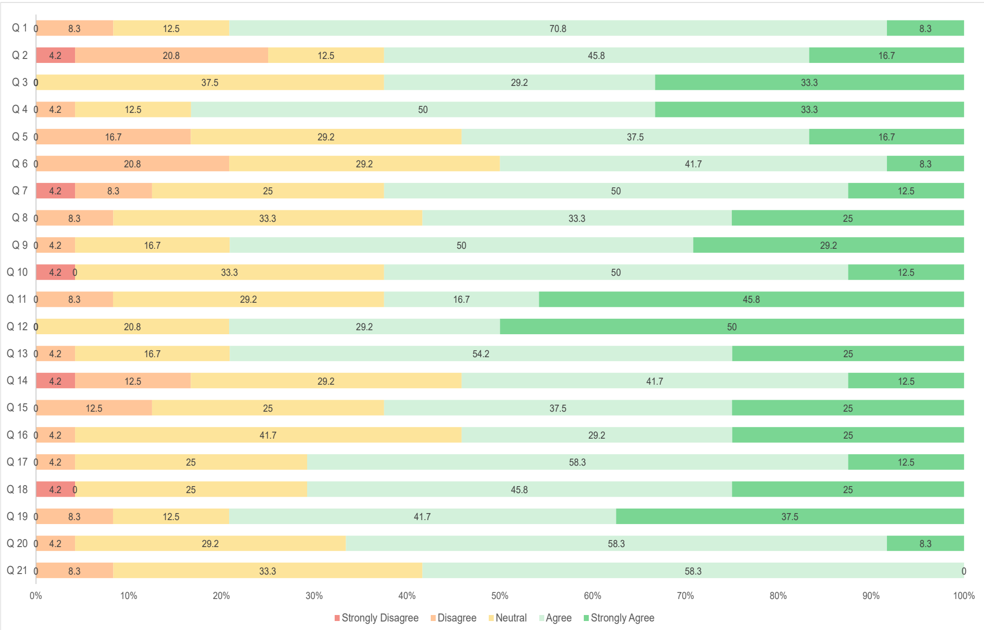

Fig.4. Break-down of the responses to the questionnaire

ARCS-related items showed high engagement. Over 80% of participants agreed that the game was enjoyable and maintained their attention (Q11, Q12). Around 70% found the content relevant and felt satisfied with their progress (Q13, Q17, Q18). Most felt confident completing the tasks (Q15), though some expressed uncertainty in interpreting complex visuals (Q16), highlighting areas for improving the interface.

Open-ended feedback highlighted the usefulness of visual aids, challenge cards, and incremental decision-making. Some participants suggested shortening the overall game duration and introducing a clearer onboarding process, especially for those unfamiliar with ML terminology.

Additional feedback emphasized how MLopoly translated abstract ML topics into compelling contextualized challenges. Many respondents appreciated how challenge cards and sector-specific tasks improved retention. Visual elements such as performance metrics and outcomes of choices were cited as particularly helpful. Some participants also noted the need for better pacing and clearer feedback during tuning tasks, suggesting concrete improvements for future iterations.

Overall, the questionnaire suggests the strong potential of MLopoly as a complementary educational tool. The combination of structured gameplay, visual feedback, and scenario-driven challenges supported both cognitive learning and motivational engagement. While some usability aspects should be refined, the study suggests that MLopoly can support ML instruction by providing a pleasant context where students can play in teams while applying and practicing with disciplinarily rigorous concepts.

5 Conclusion & Future Work

This study explored the potential of game-based learning to address the challenges of teaching foundational machine learning concepts. We introduced MLopoly, a serious game inspired by Monopoly, designed to teach key ML principles through scenario-driven gameplay and interactive decision-making.

Our findings suggest that MLopoly effectively enhances learner engagement and facilitates conceptual understanding, particularly for those with limited prior exposure to machine learning or programming. While our evaluation was limited in scale, both qualitative and quantitative feedback indicate that the game offers a compelling ancillary tool for disciplinary instruction and motivation.

The current implementation of MLopoly was limited to Decision Trees. However, thanks to the modular design, the board and mechanics can be easily reskinned to address additional ML concepts, such as Random Forests and Neural Networks, thus extending the scope of the game

In addition to broader content coverage, further future directions include increased adaptability to different learner levels (e.g. personalized feedback) and the exploration of AR or web-based versions for easier usage. A more ambitious goal is scaling the game framework to other academic domains beyond machine learning, while improving instruction quality and fun.

References

1. Voulgari, I., Zammit, M., Stouraitis, E., Liapis, A., Yannakakis, G.: Learn to Machine Learn: Designing a Game Based Approach for Teaching Machine Learning to Primary and Secondary Education Students. In: Proceedings of the 20th Annual ACM Interaction Design and Children Conference. pp. 593–598. Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3459990.3465176.
2. Zammit, M., Voulgari, I., Liapis, A., Yannakakis, G.: Learn to Machine Learn via Games in the Classroom. Frontiers in Education. 7, 913530 (2022). https://doi.org/10.3389/feduc.2022.913530.
3. Gundersen, S.W., Lampropoulos, G.: Using Serious Games and Digital Games to Improve Students' Computational Thinking and Programming Skills in K-12 Education: A Systematic Literature Review. Technologies. 13, 113 (2025). https://doi.org/10.3390/technologies13030113.
4. von Davier, T.S., Larsen, A.J.H., Van Kleek, M., Shadbolt, N.: ArtBot: An Exploration into AI's Potential for Guiding Art Analysis. In: Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems. pp. 1–11. Association for Computing Machinery, New York, NY, USA (2025). https://doi.org/10.1145/3706599.3720181.
5. Aissi, M., Grislain, C., Chetouani, M., Sigaud, O., Soulier, L., Thome, N.: VIPER: Visual Perception and Explainable Reasoning for Sequential Decision-Making. (2025). https://doi.org/10.48550/arXiv.2503.15108.
6. Schumacher, C.: Cognitive, metacognitive and motivational perspectives on learning analytics : Synthesizing self-regulated learning, assessment, and feedback with Learning Analytics, https://madoc.bib.uni-mannheim.de/53712, last accessed 2025/07/24.
7. Alam, A.: A Digital Game based Learning Approach for Effective Curriculum Transaction for Teaching-Learning of Artificial Intelligence and Machine Learning. In: 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS). pp. 69–74 (2022). https://doi.org/10.1109/ICSCDS53736.2022.9760932.
8. Nordin, N., Mohd. Nordin, N.R., Omar, W.: REV-OPOLY: A Study on Educational Board Game with Web-based Augmented Reality. AJUE. 18, 81 (2022). https://doi.org/10.24191/ajue.v18i1.17172.
9. Borromeo, N.A.: Hands-On Unity 2022 Game Development: Learn to use the latest Unity 2022 features to create your first video game in the simplest way possible. Packt Publishing Ltd (2022).
10. Gogus, A.: Bloom's Taxonomy of Learning Objectives. In: Encyclopedia of the Sciences of Learning. pp. 469–473. Springer, Boston, MA (2012). https://doi.org/10.1007/978-1-4419-1428-6_141.
11. Bloom's Taxonomy of Educational Objectives, https://teaching.charlotte.edu/services-programs/teaching-guides/course-design/blooms-educational-objectives/
12. Keller's ARCS Model | EBSCO Research Starters, https://www.ebsco.com/research-starters/social-sciences-and-humanities/kellers-arcs-model
13. SinaGholami/MLOPOLY: This is the Result and questionnaire of MLOPOLY article., https://github.com/SinaGholami/MLOPOLY