EG UK Computer Graphics & Visual Computing (2025)
Y. Sheng and A. Slingsby (Editors)

MechHead: Arcade Framework for Game Design Education

David James[1], Kieran Hicks[1], Mike Beardwood[1], and Chris Headleand[1]

[1]Staffordshire Games Institute
School of Digital, Technologies, Innovation and Business
University of Staffordshire, UK

Abstract

This technical report presents MechHead, an open-ended, arcade-style game framework developed in Unreal Engine. MechHead is used as a pedagogic tool to enables students to engage directly with key design concepts such as gameplay balance, encounter design, and level design within the University of Staffordshire games curricula. Critically, MechHead is both a pedagogic tool and an award-winning (the 2023 TIGA award for Best Arcade Game) commercial-quality game, bridging academic and industry expectations. It is designed to be inherently modular to allow learners to iterate on complex systems without being burdened by foundational boilerplate. Its architecture is informed by constructivist pedagogies, promoting reflective practice through structured experimentation.

1. Introduction

This technical report presents MechHead, a commercial quality game framework designed for education. Game design is an inherently practical discipline, where concepts must be implemented, tested, and refined [JH24]. Built in Unreal Engine, MechHead provides a modular and extensible starting point that includes core gameplay features, customisable mechanics, and scaffolding. The framework allows students to focus on specific technical games design topics using standard industry tools. Although theoretical principles such as level structure, pacing, flow, and balance can be taught conceptually, their value in a learning context is only fully operationalised through practice. We liken this to swiming, where although the process can be understood purely theoretically, one must enter the water to develop mastery.

2. Background

Traditional pedagogical approaches often isolate design theory from implementation practice, limiting the opportunities for meaningful iteration and reflection. The reflective approach argues that design knowledge emerges through cycles of implementation and review [Sch17], and this approach has been successful in other visual computing disciplines [RRJH17]. Yet this cyclical learning process is difficult to achieve if students are overburdened with building foundational boilerplate from scratch as is common with Games education.

Effective game design is not merely the production of artefacts, but a process that requires active engagement with player experience [Bea24, TZ03]. Game engines such as Unreal Engine provide powerful technical platforms, but have steep learning curves, resulting in learning experiences that are often either too tool-focused or overly reliant on reproduction. While some curricula adopt the modification of commercial games, these are often outdated experiences or the leap to implementation remains steep, time-consuming, and overfitted to that specific game. This disjunction between theory and practice is a well-discussed, but poorly documented issue in games education.

MechHead was developed to fill this gap. It provides a technically grounded, rich game framework that allows students to apply design thinking in a sandbox of interactive systems.

3. MechHead

MechHead is a Commercial Quality game developed in the Staffordshire Games Institute, and winner of the 2023 TIGA award for Best Arcade Game. It adopts top-down gameplay where the player controls a robot (a 'Mech') using standard keyboard and mouse inputs. Core gameplay elements are already in place, allowing students to concentrate on designing mechanics, encounters, and levels.

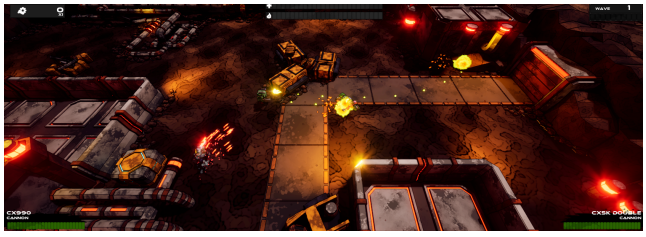

Figure 1: *MechHead Gameplay*

Figure 2: *MechHead 'Mech Configuration' tool*

A central feature of MechHead is its weapon system. The player mech can equip two weapons simultaneously, with default samples provided to demonstrate functionality. These weapons are fully modular and expose key gameplay variables through Unreal's Blueprint interface. Students can adjust parameters such as fire rate, ammunition consumption, recoil intensity, projectile speed, damage output, and visual effects. This allows for rapid experimentation and iteration on game feel and balance.

The framework also includes a suite of enemy types, each with basic artificial intelligence behaviours. These enemies operate within defined AI states (Idle and Combat) and support behaviours such as patrolling, chasing the player, ranged attacks, and fleeing. Students can assign behaviours through custom menus and modify variables such as health, speed, and attack type. The modular enemies approach encourages experimentation with encounter design, and artificial intelegence.

To support level creation, MechHead includes a range of prebuilt assets, including modular meshes, decorative materials, particle effects (e.g. fire, smoke, wind), and audio cues. Key game logic elements, such as checkpoints and level goals, are pre-integrated to enable progression tracking and gameplay flow. Upon reaching the goal, a summary screen displays the player's time, death count, and performance score.

4. Design Considerations

The design of MechHead was guided by three core principes

- *Abstraction*: To allow isolation of specific learning all gameplay systems within MechHead allow isolated modification through exposed variables and editable components.
- *Clarity*: To facilitate independent learning all components within the system such as Assets and Blueprints are meticulously labeled, categorised, and commented to facilitate independent exploration and learning.
- *Flexibility*: To enable creative freedom and support creativity, all in-game components are modular, and easily exchanged for alternative objects.

This required balancing complexity and accessibility, which inspired the design choice of an arcade game as a model. This offered students sufficient depth to meaningfully experiment and iterate, without overwhelming them with excessive control schema.

5. Pedagogic Applications

MechHead allows students to explore game mechanics and balance via guided experimentation. Variables exposed through the unreal engine can be adjusted in real time, letting students prototype gameplay variations quickly with limited technical knowlege. Importantly, this approach doesn't limit more advanced development or customisation as the students have full access to the engine (often not available through other approaches - such as level editing systems like 'Hammer'.

The framework introduces basic scripting through Blueprint visual programming. After manipulating variables, students iteratively progress to more advanced features. This structured approach helps build confidence and transition naturally from parameter tuning to logic construction, always linked to observable goals.

Enemy AI in MechHead operates on a dual-state system (Idle and Combat), with behaviors such as patrolling, and fleeing [R*99]. These can be combined to create varied encounters that teach pacing, challenge curves, and encounter design.

Level construction supports spatial and experiential design. Built-in checkpoints and customizable visual assets let students shape both function and atmosphere, promoting environmental storytelling and integrated design thinking.

6. Conclusion

MechHead represents an industry recognised and pedagogically grounded solution to a persistent challenge in games education. By providing a modular, documented, and expandable framework, it allows learners to focus on creative problem-solving, gameplay iteration, and reflective design practice. Its structured yet flexible architecture supports a range of learning outcomes within an commerical quality (and award winnng) game.

Using MecHead in our curricula has led to improved student satisfaction, and higher student atainment. Future work will focus on analysing the pedagogic value of the Games Framework approach, with MechHead being leveraged as a research vehicle.

References

[Bea24] BEARDWOOD M.: *Fundamental Level Design and Analysis: A to B*. CRC Press, 2024. 1

[JH24] JAMES D., HEADLEAND C. J.: Games design frameworks: A novel approach for games design pedagogy. In *Proceedings of the 18th European Conference on Games Based Learning* (2024), Academic Conferences and publishing limited. 1

[R*99] REYNOLDS C. W., ET AL.: Steering behaviors for autonomous characters. In *Game developers conference* (1999), vol. 1999, Citeseer, pp. 763–782. 2

[RRJH17] ROBERTS J. C., RITSOS P. D., JACKSON J., HEADLEAND C.: The explanatory visualization framework: An active learning framework for teaching creative. *IEEE Transactions on Visualization and Computer Graphics 10* (2017). 1

[Sch17] SCHÖN D. A.: *The reflective practitioner: How professionals think in action*. Routledge, 2017. 1

[TZ03] TEKINBAS K. S., ZIMMERMAN E.: *Rules of play: Game design fundamentals*. MIT press, 2003. 1