

CSE 493V Final Project Report

Introducing CHESS EN GARDE

JULIA WANG and NOAH KROHNGOLD, University of Washington



Fig. 1. A John Wick-style duel in Chess En Garde that takes chess to a new level. When pieces are captured, the two involved pieces are launched into a vigorous duel sequence where they fight to the death. Whichever piece ends up hitting the other successfully is the one that completes the capture.

Deciding between games to play is never a fun time. What if you could get the best of both strategy and adrenaline-boost shooters? Power up your Oculus and experience the thrill of dueling an opponent by wielding a weapon at arms-length – all while plotting the checkmate of the other King.

1 INTRODUCTION

FPSCchess started as a student project from DigiPen that combines the age-old strategy game of chess with the adrenaline of first-person shooters in a PC game. We sought to draft that same complexity of gameplay together with the immersion of true first-person that VR brings to the table of gaming, and make both chess dueling an option for all.

Introducing Chess En Garde!

1.1 Contributions

- Implemented rigidbody physics and interactables for the Oculus Quest 2 system using model prefabs and the XR Interaction Toolkit.
- Implemented auto-move and auto-shoot system for the black player for bot engagement in both the chess and duel scenes.
- Built organized, clean scripts for future continued development on the game to easily fit into the existing codebase.

Authors' address: Julia Wang, julialwa@cs.washington.edu; Noah Krohngold, krown@cs.washington.edu, University of Washington.

- Implemented player positional tracking, snap-to-grid, and state saving when transition between scenes / game modes.
- Started the process of engineering quality-of-life through transitions, control choice, and on-screen prompts.

2 METHOD

Chess En Garde's goal is to replicate the novelty that FPSCchess introduced and bring that concept to virtual reality platforms. This involved creating environments for traditional chess gameplay as well as combat between pieces involved in a capture. Pieces that lose a duel are removed from play. Models can be pulled from the Unity Asset Store for simplicity. Information on when it's the player's turn, what pieces are no longer in play, whether or not the player is in combat, and what piece the player is attempting to capture must be tracked throughout the game, as this defines how core mechanics interact with each other in spawning pieces and representing victory during duels. For demo purposes, we also included a few preset moves that the black pieces would make. When playing chess, only the player's pieces are interactable and during the combat phase only the weapon is interactable, where the phases use a ray and direct interactor, respectively.

3 IMPLEMENTATION DETAILS

We built Chess En Garde using the Oculus Quest 2 platform. The Quest 2's feature set lends itself nicely to the game development experience as the controller's features and reliability streamlined the development process. The game itself was built using Unity, relying on assets from Unity's asset store for all of the models we needed. Our first priority was dedicating spaces to each scene we would need, choosing the assets we wanted, and placing them within each necessary scene to familiarize ourselves with Unity as a development platform.

Once the chessboard and duel scenes were staged correctly, we pivoted entirely to the game logic necessary for the minimum viable product. Utilizing the XR Interaction Toolkit for interacting with objects in the scene and creating the correct combination of mesh and rigidbody components was necessary for the physics of the scene in which objects can be thrown and collide with others.

Logic particular to the chessboard scene largely revolved around ensuring that the chessboard remained the focal point of interactions. Chess pieces snap to squares on the board to maintain precise placement, and player pieces were only interactable when it was the player's turn while the enemy pieces were never interactable, only registering collisions for when duels were initiated.

The duel scene introduced a firearm that the player could interact with. Following the format of a traditional pistol duel, we limited players to only one bullet during combat. Before permitting shots to register on either the player or enemy, the player must "holster" their weapon, meaning that they hold the weapon pointed directly downward until a countdown finished, thus instructing the player to shoot. The countdown appeared as a UI text element directly above the enemy the player is fighting. Hit-detection logic involved creating a physics raycast forward from the location of the gun barrel. When pulling the trigger, a bullet model is instantiated from the barrel location and a large amount of force is applied in the forward direction, thus propelling the bullet. Since only one bullet is fired, we can just track the tag of the object that the bullet collides with to determine if the enemy was struck by the bullet. If the player successfully shot the enemy, then the game transitioned back to the chessboard scene, in which the killed piece is now removed.

4 EVALUATION OF RESULTS

Compared to FPSChess, Chess En Garde is not as developed or complex in terms of movement mechanics, game design, or environment aesthetics (yet). Considerations to be made in comparing these two products include quality of life (motion sickness from VR), a more dynamic duel scene with different types of guns or accuracy etc. We intend on continuing this development after the class concludes and we both graduate, so we will further enhance and differentiate our product moving forward!

5 DISCUSSION OF BENEFITS AND LIMITATIONS

We handed our application to friends with no prior explanation and observed what they picked up immediately and what they struggled with or didn't like. We had time to fix some of our noticed bugs (green), but saved others for outside the scope of this short project

time (orange). Findings include:

Positive (+):

- (1) Movement and rotation player mechanics is intuitive
- (2) Transitions prompt clearly what the next step is
- (3) Concept is enjoyable and duelling is fun overall

Negative (-):

- (1) Tutorial that sets up the context starting into the game
- (2) "Reel-in" feature to adjust distance of piece grab
- (3) Black and white pieces are both grabbable, which confuses player role
- (4) Duel mode should introduce hand models to make it easier to grab gun
- (5) Pieces should not be able to accidentally fall off / get through off the board

6 FUTURE WORK

During the live demo and development process, we put together and maintained a (nonexhaustive) list of bugs / features that we intend on fixing in the future, from most straightforward to most involved:

- (1) GAME OVER and GAME START screens for game flow
- (2) Prevent other white pieces from being knocked over by the current move
- (3) Player mechanics for rotation / "reeling in" using Oculus controller
- (4) Selection-based chess piece movement instead of drag-based
- (5) Save game state before / after duel so pieces don't revert
- (6) Actually enforce the rules of chess instead of operating on the honor principle
- (7) Bot moves are currently not randomized and are quite limited to reduce the volume of hard-coded numbers in the codebase
- (8) When the player "kills" a piece that is hard-coded, all of the black pieces disappear instead of just the one that was killed
- (9) 2-player mode (ideal)

7 CONCLUSION

We look forward to continuing to work on our demo of Chess En Garde, and build it out to be a cleaner, smoother, and more engaging 3D chess duel game. Obviously there's a lot of bugs, but we spent most of our time learning & putting together an organized basis to continue development moving forward. Maybe you'll see us in the Oculus store sometime after this summer! :)

ACKNOWLEDGMENTS

Thank you to Doug and the rest of the teaching team for a fantastic quarter, and John Akers for consistently being a timely and willing resource to any issues we ran into throughout the project. We also appreciate our test players and everyone who came to our poster session and live demo!