8-2024

Lessons Learned Over a Growing Development Cycle in Medical Simulation

Alex Pacheco-San Martin
East Tennessee State University

Lessons Learned Over a Growing Development Cycle in Medical Simulation

---

A Culminating Experience

presented to the faculty of the Department of Digital Media

East Tennessee State University

In partial fulfillment of the requirements

For the degree

Master of Fine Arts in Digital Media

---

by

"Alex" Enrique Pacheco San Martin

August 2024

Todd Emma, Chair ____________________

Gregory Marlow ____________________

Jacy Richardson ____________________

Abstract

Lessons Learned Over a Growing Development Cycle in Medical Simulation

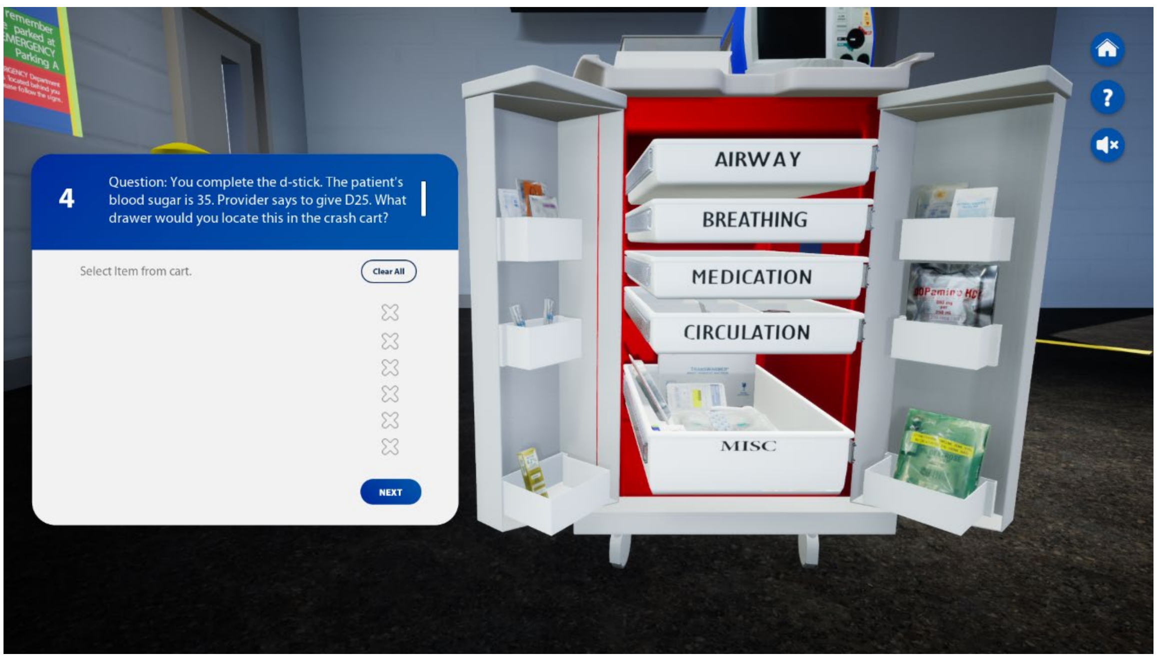
By

"Alex" Enrique Pacheco

For the last five years, East Tennessee State University's Department of Digital Media has partnered with the East Tennessee Children's Hospital in downtown Knoxville to create a collection of training modules for their nursing staff. Dozens of students have devoted thousands of hours both inside and outside of the department's various production courses to build the eleven modules currently in use today. However, development has not been without its hurdles, particularly in moving from one project to the next. Poor documentation and miscommunications compounded with faults in the projects' design philosophies to repeatedly delay production. While each deliverable was playable by the end of the given semesters, there were often bugs and other quality issues left unaddressed. Luckily, graduate assistants and interns were brought on to help direct students in class, and to finish the projects on off-semesters. As one of those graduate assistants, I've had the pleasure of working alongside Stephanie Nicely, our project manager and hospital liaison to build on the works of past developers, assist in managing the development of current projects, and lay groundwork for future growth.

Chapter 1 Introduction.

In 2019, Taylor Oglesby reached out to Todd Emma about the development of potential training software for emergency response. Students in the year-fall/spring production course then developed the Crash-cart modules, a collection of three multiple-choice tests for response crew members to be played through on PC. The tests are accompanied by an interactive digital representation of the hospital's emergency response Crash-cart, which participants must interact with to locate the appropriate tools for treating the scenarios posed by the questions.



The final deliverable was produced in Unreal Engine 4.23 based on reference photography from the hospital and content written by the East Tennessee Children's Hospital's (ETCH) Organizational Development and Learning Department (ODL). Following the success of the Crash-cart modules, the development of the de-escalation scenarios (De-Esc) was proposed.

Like the Crash-cart modules, De-Esc was comprised of a series of test questions intended to guide learners through de-escalating stressful scenarios with patients and their families.

However, unlike the previous project, De-Esc was developed for virtual reality on the Oculus Rift. Each scenario takes place in a virtual replica of the hospital, complete with non-player characters (NPCs) that react to the learner's actions and responses. In doing so, the team hoped to immerse the nurses in training in the environment and keep them engaged with the material. It should be noted though that while the virtual environment was itself designed to engage the learner, the questions themselves were still presented in the form of a multiple-choice quiz, on a hovering user interface (UI) panel.

The most recently completed module collection is designed to lead nurses through maintenance and replacement of a tracheostomy catheter, and is dubbed Trach & Vent or T&V. This project serves as a supplement to training performed on physical mannequins and is our collaboration's first attempt at a fully immersive VR experience. Much of De-Esc's functional and visual infrastructure was rebuilt to support an interactive environment for the learner. Rather than simply answering questions, players instead select, grab, and utilize virtual objects to treat the simulated patient. Guidance is provided by in-game voice overs, visuals, and occasional outside assistance from ODL staff. The development of Trach & Vent was completed in May of 2024, concurrently to the previsualization of the team's next project.

I joined the ETCH collaboration halfway through the development of De-Esc, in Spring of 2022. During the semester's production class, I served as one of the two programmers on the project, as well as the asset development team lead, managing the quality of models and textures being implemented in the project. The team's deliverable at the end of that semester didn't meet my personal expectations as a developer, and I chose to remain on the project through the Summer and subsequent Fall to polish the project. I worked alongside Stephanie Nicely, our new project manager and contact at ODL, to update both the De-Esc and Crash-cart projects, before continuing as the lead programmer on Trach & Vent. Throughout my involvement, I have built and improved upon the code written by the collaboration's past developers, helped mediate between the content authors at ODL and our development teams, and helped shift our design philosophy and practices to ease further modification of our projects with the intent of improving our production courses' development cycle.

Chapter 2 Review of Literature

In the medical field, simulation-based training (SBT) traditionally refers to the use of educational stand-ins for practice prior to live patient treatment, and has been used clinically for thousands of years. Clay models of livers dated between 1900 and 1600 BCE were used in ancient Babylonia to forecast the outcomes of illnesses (Owen, 2016). Aristotle and Hippocrates suggested that training models eased repetition, and that feedback and guidance were necessary for developing expertise (Owen, 2016). From clay to bronze to ivory, to wood, wax, and plastic, mannequins evolved considerably through to the 20^{th} century, when more lifelike elements were implemented into the designs, such as artificial breath, blood pressure, and eye motion.

The 1980s saw the development of high-fidelity simulation (HFS) environments (Bienstock & Heuer, 2022) defined by two main characteristics: fidelity and modality. Modality "refers to the type(s) of simulation equipment… such as a task trainer, standardized or simulated patient (SP), full-body mannequin, or screen-based simulation" (Carey & Rossler, 2024). Fidelity meanwhile refers to the degree of realism and believability created by the setting, equipment, and scenario. These HFS environments were the precursors to modern clinical simulations which are "in the process of moving to hybrid models that incorporate mannequins or actors while using high-fidelity virtual reality (VR) models and augmented reality (AR)" (Bienstock & Heuer, 2022).

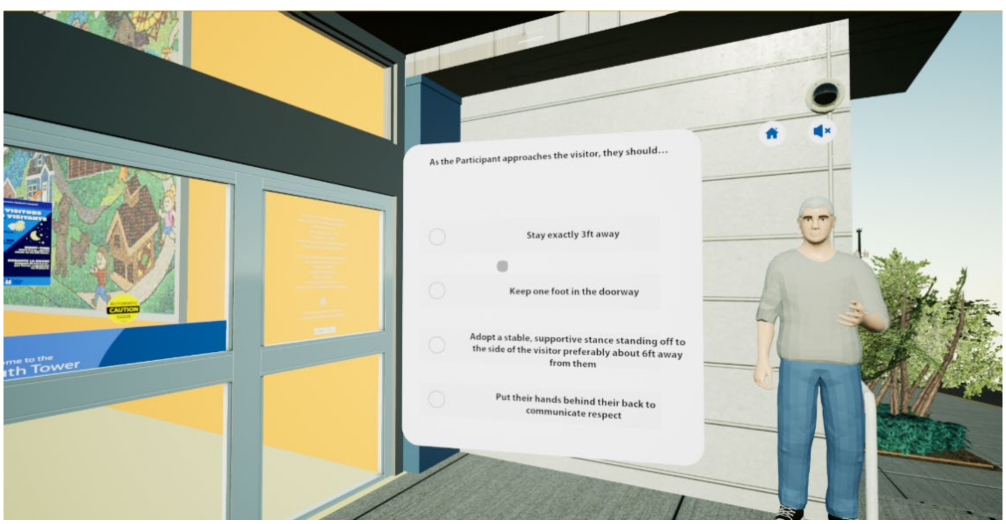
Since its emergence in the 1960s, virtual reality hardware developed parallel, but separate to the development of clinical simulation. VR hardware primarily consists of a head-mounted display capable of three-dimensional tracking, the first of which was developed in 1968 by Ivan Sutherland (Pillai & Ismail, n.d.). Early versions of these headsets were often held back by requirements, restrictions, and presentation. *Virtuality* attempted to introduce VR to arcades in 1991 with multiplayer virtual reality “pods”, which were intended to make gameplay a spectacle for outside observers, rather than entertaining for the players (Fowle, 2015). In the 80s and 90s, both Sega and Nintendo developed their own VR gaming systems but saw limited to no success. It wasn’t until 2015 when Oculus and HTC Vive would introduce high-quality VR headsets to the consumer market.

This consumer-friendly VR technology was initially experimented with by indie studios, resulting in graphically simple games with compelling interactive gameplay such as *Superhot VR* and *Job Simulator*. These and other titles focused on making the player’s actions directly impact the world around them, bringing the worlds “to life” despite their stylized visuals. Interactive VR environments like these have been greatly sought after in the medical space ever since, because of their low physical resource cost, and the ability to develop lifelike training scenarios. The focus in such scenarios is on decision making, critical thinking and clinical reasoning, with scenarios being designed to replicate human interaction in the real world (Pottle, 2019).

These interactive scenarios are an example of game-based learning (GBL), in which games are used to enhance learning and teaching (Wiggins, 2016). GBL works in concert with gamification, the integration of game-design elements into non-game contexts (Krishnamurthy, 2022). Both terms are new in the field of education, having only gained traction over the last few decades, but tied in well to the educational philosophy of medical simulation. Simulations also provide low-risk learning environments that allow for repetitive training without endangering actual patients at the hands of inexperienced learners. By integrating game mechanics, simulations enhance engagement, motivation, and retention of knowledge among learners.

Chapter 3 Research Design & Methods

As mentioned previously, my involvement with the ETCH collaboration began in January 2022, as part of the Spring semester's production class developing the de-escalation scenarios. Two of the four planned scenarios for the project had been completed during a previous semester, and the Spring 2022 production team aimed to create the third and fourth scenarios to finalize the project. The team was managed by faculty overseers Tod Emma and Jacy Richardson, with Kimberly Campbell as our contact at ODL. The team was directly led by two graduate assistants (GAs), Grayson Wiles and Stephen Overton, with Megan Smith as the lead animator and Maggie Shelton as the lead environment artist. Initially, I joined the team to oversee the quality of the undergraduate modelers' work, aiming to reduce the workload on the GAs. However, due to my versatile skill set, I soon assumed multiple roles within the team.



The project's previous lead programmer, Caleb Waldrop, had moved on from the project after building the initial scenarios. The responsibility of development then passed to Seth Bowen from the computer science department. However, since the project was expected to be expanded upon by future classes and most digital media department scripters lacked expertise in C++, it was decided that development would continue exclusively in Unreal’s visual scripting interface, Blueprint. Bowen unfortunately lacked experience using visual scripting, so I became his mentor and helped integrate his code into the project.

As I was already assisting our programmer while managing our modelers’ consistency, I was consequently tasked with importing the various animations, models, textures, and other assets into the final project. I collaborated closely with Shelton and Overton to match the reference footage from the hospital with a cohesive visual style. Smith and I worked to integrate her team’s animations into the project, arranging them into a behavior tree for the project’s NPCs.

At the end of the Spring semester, the team had succeeded in delivering a playable executable, and Wiles and Overton left the project. However, there were gameplay errors, missing animations, and spelling mistakes affecting the final quality. I chose to remain on the project following the conclusion of the semester and worked that Summer and Fall to polish the De-Esc scenarios with Stephanie Nicely. Together we refined both the De-Esc and Crash-cart scenarios, by both updating content and adding quality of life features for the players, including the ability to zoom on previously illegible text elements.

While completing these projects, Samuel Pinnex and Levi Stone Walker led the pre-visualization of Trach & Vent as our collaboration's first dedicated interns. This transitional period highlighted several workflow issues within the production classes. Each ETCH project has been built on the infrastructure of its predecessors, which at a glance has a few advantages. The projects all take place in the same effective space, and the realistic style and asset complexity have been predominantly consistent throughout. Further, from an intuitive perspective, much of the code felt transmissible between projects. The Crash-cart project featured a large UI widget system for its menus and test system, which was modified to receive VR controller input for the De-Esc scenarios. It followed that such a system could be further modified for Trach & Vent's object manipulation system. However, the Crash-cart, De-Esc, and Trach & Vent projects were all fundamentally different, and only shared superficial similarities.

The misguided attempt to reutilize old code rather than rebuild particular systems was further hindered by a flaw in each project's design philosophy. The projects were not designed to be built upon, as students primarily focused on meeting guidelines and deadlines. The nature of the semester-based production classes potentially hindered the project's development beyond the classroom. For example, the restriction to strictly use Blueprint for the project's code to ease augmentation in the future was followed, but the actual script was not written with readability and modification in mind. As a result, editing and repurposing code required inefficient back and forth with the previous projects' programmers.

Examples of these moments include comments in the code for the main player character in scenarios one and two of De-Esc:

> "This fiddly-looking block handles both the widget interaction and switching the Widget Interaction Component's hand (not really, it just activates for one hand and deactivates for the other)."
>
> "Doesn't actually do anything, but when I try to delete it, the engine says that there are 2 memory references so I'm too scared to."

The first comment is a poor descriptor of the underlying code and does not explain its functionality. The second comment highlights the clutter within the project due to the experimental nature of VR development and the tendency to build on mistakes rather than fix them due to time constraints.

Larger documentation issues included the lack of written records for packaging deliverables for the hospital. The process was only known to those in charge and those testing incremental builds. Through private Discord messages, I discovered our current packaging method was based on a video by Asbjørn Thirslund (Brackeys), a game developer using Unity. Our team had modified his method for use with Unreal Engine, but our precise instructions were not recorded until March 2023.

I made the decision to develop Trach & Vent as a facsimile of De-Esc rather than a modification or extension. While some assets were copied, the project was developed in Unreal 5, requiring changes in visual assets and a rebuild of the underlying code. Old systems were purged and reimplemented to reduce bloat before new systems were built on the refurbished foundations. The team began designing with iteration in mind rather than simply creating a functional product.

Despite the redesign, Trach & Vent faced development hurdles. Creating an interactive virtual world was more complex than anticipated, causing production to be split into two parts: the first half completed in Spring 2023, and the second in Spring 2024. Misunderstanding the project's scope led to delays, compounded by three common issues:

Communication with Content Authors: Each clinician's experience is marked by personal tricks and quirks for performing necessary procedures learned throughout their medical careers. Because their experiences and methods vary slightly, feedback from ODL and the hospital's clinical staff was occasionally self-contradictory. To remedy this, Nicely and I would compare various requests, compare them to the original project script, and try to mediate a generalized procedure for use in Trach and Vent.

Software Installation Delays: Due to the department's security features, students were unable to permanently install necessary software on laboratory computers. Each class began with reinstalling GitHub, Meta Quest Link, and other SDKs, cloning the project repository, and compiling shaders. This process repeated every class, causing significant time loss.

Dissonant Design Philosophy: Programmers and designers often had conflicting approaches. In-house tests focused on finding bugs and glitches in edge cases, sometimes losing sight of the project's intent and audience. While beneficial for sanitizing input, too much time was spent on perfecting minutiae rather than consistently polishing mechanics.

Chapter 4 Results

Each of the previously mentioned issues are still present in the ETCH production teams today, but efforts have been made to iron out as many bumps as possible. The largest change is an integration of a modular design philosophy into the management structure. Samuel Pinnex and Levi Walker both worked on the De-Esc production team in some capacity, and they will continue working with ETCH through till their graduation from the Digital Media Master of Fine Arts. As the first students through our new internship track, they have established a pipeline for undergraduate juniors and seniors in the digital media and computer science programs to work on ODL's projects and continue sharing their expertise through their graduate career. Some students in this pipeline are expected to work for a few years on similar projects and are incentivized to make things easier on themselves in the future by making adaptable design decisions in the present. These students also work together alongside faculty to recruit their successors in the program, which has thus far enabled a sustainable trickle of computer science students to assist our programming teams. Going forward, as we lose graduating members of the production and management teams, it is expected for us to leave behind a collection of instructions and guides for modifying the previous projects to minimize information loss.

An attempt is also being made to streamline the production process. Our suite of software is being added to the in-house computer system image, which should prevent further in-class delays. In addition, the team has stopped packaging iterative test builds for the hospital. Rather than spend hours packaging bi-weekly alphas for testing and feedback, I have worked alongside Nicely and trained her to access our projects through GitHub and Unreal and test them from the engine herself. With this change, she now has constant access to the current production build and the most recent stable build and can test our progress at any time. Content authors will now also be brought in sparsely towards the end of production for final adjustments, rather than script rewrites mid semester.

The team has also shifted its design philosophy to a true gamification of the material with its upcoming non-VR project. The currently unnamed game hopes to convert the hospital's library of training case studies into a puzzle and deduction style game, where nurses will interact with a patient and their paperwork to diagnose and escalate them as necessary. While a proof of concept is still in production, the intent is to host the game on a website where nursing staff can compare their times, in the hopes a leaderboard will incentivize repetition and improvement. The lead programmer on the project, Caden Lafollette, also began development of the project with the intent of reading case study information directly from the hospital's library, rather than hard coding scenarios into the game. This functionality would allow the content authors to easily update the game as they see fit by simply editing their existing systems. While importing scenario information from a text document has always been a consideration in previous projects, such a framework was never implemented due to conflicts with pre-existing systems and time constraints. As this new project is largely independent from those previous, we hope such an integration will be possible.

Chapter 5 Conclusions

For all the time and effort that I've devoted to our collaboration with ETCH, I've been rewarded with unforgettable experiences. Working with our various development teams, ODL, and Stephanie Nicely has provided me a skillset in management that I didn't expect to learn going into the project. I hope that my work will continue to be built upon, and that future production teams won't hesitate to reach out if they need assistance. We have taken a few steps towards easing future development, but I expect that future development in VR will continue to be a bumpy road.

Parallel to our collaboration with the East Tennessee Children's Hospital, there have been seven versions of Unreal Engine, five versions of Autodesk Maya, and five headsets developed by Quest, formerly Oculus. These changes in hardware and software are inevitable, but our skillsets have improved in tandem. I believe that as we continue developing projects for the hospital, we should consider what technologies we can best utilize to achieve our goals. In particular, future developments in VR may allow for more detailed controls using gloves and haptic technologies to more accurately mimic real world object interaction. This is also an area that would expand our partnership with the computer science department, were we to develop our own hardware. We should also keep an eye out for emergent technologies that may be applicable towards expanding the scope of our projects; for example the work of Austin Caskie (Caskie, 2023), a Master's student from North Carolina State University, is the first step towards implementation of complex wire, rope, and thread in a VR setting, and could potentially be used for training stitching or suture once the precision can be increased.

A great deal of clinical practice is based on rote learning, the memorization and repetition of action until behavior becomes habitual. This is good as repetition improves competency, but bad in the sense that individuals develop their own methods and habits. As mentioned before, various clinicians at the children's hospital had conflicting beliefs on the methods and practices that should have been followed in Trach & Vent, which Nicely and I had to mediate into a generalized practice. I expect this will occur repeatedly over the course of future projects. To mitigate the impact of these conflicts, I have found that it's important to rely on our agency as developers. We have a finite amount of time to build these projects, and we can't afford many delays. Development should not pause until a consensus is reached among the content authors, but rather continue along the predetermined script, and then be modified once a consensus is reached.

Further, it is important to understand that our client is currently inexperienced in the technical aspects of the game design profession. In my experience, it is best to build a mechanic to completion so it can be presented to staff and adjusted, rather than ask how it should be designed to begin with. This gives us agency as designers, and helps the authors visualize our intent. When producing these mechanics our programmers should create accurate and detailed documentation, rather than focus on bug fixing. We should not be accounting for players turning their controllers upside-down unless that is an intended action. The nursing crew is not attempting to glitch through the game, they are attempting to learn, and the only cases to focus debugging are those that occur most commonly during standard play of the scenarios.

Until now, each project has had at most three programmers, which seems to be an upper limit for Blueprint based development. Unfortunately, Unreal's source control has been in beta for a few years now, and has caused many issues during concurrent script editing, limiting the amount of code any given team member can write at a time. Hopefully with future versions of the engine and an increase in C++ literacy in the program, new methods to control and script the projects will increase our efficiency. I want to thank the programmers and scripters before me, Alexander Sedlak and Caleb Waldrop, for establishing the groundwork on which the ETCH projects were built.

I would like to extend that thanks to every student who's touched the ETCH projects however briefly, and to the faculty whose support we couldn't do without. A further thanks to the hospital staff, both clinical and administrative, who provided the opportunity to work with them and been patient through all our production hiccups. I hope that our collaboration continues into the foreseeable future, and that our work continues to provide fun and interactive tools to aid the nursing staff, and to help the families of those in need.

References

Bienstock, J., & Heuer, A. (2022, June 24). A review on the evolution of simulation-based training to help build a safer future. *Medicine, 101*(25).

Brackeys. (2017, April 12). *How to build / export your game in Unity (Windows | Mac | WebGL)*. Retrieved from YouTube: https://www.youtube.com/watch?v=7nxKAtxGSn8&ab_channel=Brackeys

Carey, J. M., & Rossler, K. (2024). The how when why of high fidelity simulation. In S. Publishing, *StatPearls [Internet].* Retrieved from https://www.ncbi.nlm.nih.gov/books/NBK559313/

Caskie, A. (2023, April 17). *Tying knots in virtual reality demo*. Retrieved from YouTube: https://www.youtube.com/watch?v=esXcPrnosjM&ab_channel=AustinCaskie

Fowle, K. (2015, February 4). The vanguarf of virtual reality: an embarrassing arcade game. *The Atlantic*.

Krishnamurthy, K. S. (2022). Benefits of gamification in medical education. *Clinical Anatomy*, 795-807.

Mayer, R. E. (2002, November 1). Rote Versus Meaningful Learning. *Theory Into Practice, 41*(4), 226-232.

Owen, H. (2016). *Simulation in healthcare education an extensive history.* Cham: Springer International.

Pillai, J. S., & Ismail, A. (n.d.). *Virtual reality: introduction*. Retrieved from D'source: https://www.dsource.in/course/virtual-reality-introduction

Pottle, J. (2019). Virtual Reality and the transformation of medical education. *Future Healthcare Journal*, 181-185. doi:10.7861/fhj.2019-0036

Wiggins, B. E. (2016). An overview and study on the use of games, simulations, and gamification in higher Education. *International Journal of Game-Based Learning*, 18-29.