

DIGITAL GAME DEVELOPMENT STANDARDS



This document was prepared by:

Office of Career Readiness, Adult Learning, and Education Options
Nevada Department of Education
755 N. Roop Street, Suite 201
Carson City, NV 89701

www.doe.nv.gov

Draft for review by the Nevada State Board of Education on
April 15, 2021

The Nevada Department of Education does not discriminate on the basis of race, color, religion, national origin, sex, disability, sexual orientation, gender identity or expression, or age in its programs and activities and provides equal access to the Boy Scouts and other designated youth groups.

For inquiries, contact the Equity Coordinator at (775) 687-9200.

NEVADA STATE BOARD OF EDUCATION

Felicia Ortiz	President
Mark Newburn	Vice President
René Cantú.....	Member
Katie Coombs	Member
Katherine Dockweiler.....	Member
Tim Hughes	Member
Tamara Hudson.....	Member
Cathy McAadoo	Member
Mike Walker.....	Member
Wayne Workman	Member
Alex Gallegos.....	Student Representative

NEVADA DEPARTMENT OF EDUCATION

Jhone M. Ebert
Superintendent of Public Instruction

Craig Statucki, Director
Office of Career Readiness, Adult Learning, and Education Options

VISION

All Nevadans ready for success in the 21st century

MISSION

To improve student achievement and educator effectiveness by ensuring opportunities, facilitating learning, and promoting excellence



TABLE OF CONTENTS

Nevada State Board of Education / Nevada Department of Education iii

Acknowledgements / Standards Development Members / Business and Industry Validation /
Project Coordinatorvii

Introductionix

Content Standard 1.0 Context of Digital Game Development..... 1

Content Standard 2.0 Foundations of Game Design and Development..... 2

Content Standard 3.0 Game Design..... 3

Content Standard 4.0 Programming for Digital Game Development 4

Content Standard 5.0 Build a Game..... 5

Content Standard 6.0 Legal and Ethical Issues in Game Design and Development..... 7

Content Standard 7.0 Advanced Topics in Digital Game Development..... 8

Crosswalks and Alignments 9

ACKNOWLEDGEMENTS

The development of Nevada career and technical standards and assessments is a collaborative effort sponsored by the Office of Career Readiness, Adult Learning, and Education Options at the Department of Education and the Career and Technical Education Consortium of States. The Department of Education relies on teachers and industry representatives who have the technical expertise and teaching experience to develop standards and performance indicators that truly measure student skill attainment. Most important, however, is recognition of the time, expertise and great diligence provided by the writing team members in developing the career and technical standards for Digital Game Development.

STANDARDS DEVELOPMENT MEMBERS

James Black	Teacher	Desert Pines High School
William Bumgardner	Teacher	Dayton High School
Kristina Erb	Teacher	Douglas High School
Monique Gaudin	Project Facilitator	Clark County School District
Daniel Hyden	Teacher	Douglas High School
Julian Jackson	Teacher	Advanced Technologies Academy
Brian Johnson	Teacher	Doral Academy, Red Rock
Kevin Mess	Professor	College of Southern Nevada
Conner Torres	Executive Producer	Dotware Games
Michael Edmonds	Education Consultant	Epic Games, Inc.

BUSINESS AND INDUSTRY VALIDATION

All CTE standards developed through the Nevada Department of Education are validated by business and industry through one or more of the following processes: (1) the standards are developed by a team consisting of business and industry representatives; or (2) a separate review panel was coordinated with industry experts to ensure the standards include the proper content; or (3) the adoption of nationally-recognized standards endorsed by business and industry.

The Digital Game Development standards were validated through active participation of business and industry representatives on the development team.

PROJECT COORDINATOR

Melissa Scott, Assistant Director
 Work-based Learning / Business and Marketing Education / Information and Media Technologies
 Office of Career Readiness, Adult Learning, and Education Options
 Nevada Department of Education

INTRODUCTION

The standards in this document are designed to clearly state what the student should know and be able to do upon completion of an advanced high school Digital Game Development program. These standards are designed for a two-credit course sequence that prepares the student for a technical assessment directly aligned to the standards.

These exit-level standards are designed for the student to complete all standards through their completion of a program of study. These standards are intended to guide curriculum objectives for a program of study.

The standards are organized as follows:

- **Content Standards** are general statements that identify major areas of knowledge, understanding, and the skills students are expected to learn in key subject and career areas by the end of the program.
- **Performance Standards** follow each content standard. Performance standards identify the more specific components of each content standard and define the expected abilities of students within each content standard.
- **Performance Indicators** are very specific criteria statements for determining whether a student meets the performance standard. Performance indicators may also be used as learning outcomes, which teachers can identify as they plan their program learning objectives.

The crosswalk and alignment section of the document shows where the performance indicators support the Nevada Academic Content Standards. Where correlation with an academic content standard exists, students in the Digital Game Development program perform learning activities that support, either directly or indirectly, achievement of the academic content standards that are listed.

All students are encouraged to participate in the career and technical student organization (CTSO) that relates to the Digital Game Development program. CTSOs are co-curricular national organizations that directly reinforce learning in the CTE classroom through curriculum resources, competitive events, and leadership development. CTSOs provide students the ability to apply academic and technical knowledge, develop communication and teamwork skills, and cultivate leadership skills to ensure college and career readiness.

The Employability Skills for Career Readiness identify the “soft skills” needed to be successful in all careers and must be taught as an integrated component of all CTE course sequences. These standards are available in a separate document.

The **Standards Reference Code** is only used to identify or align performance indicators listed in the standards to daily lesson plans, curriculum documents, or national standards. The Standards Reference Code is an abbreviated name for the program, and the content standard, performance standard, and performance indicator are referenced in the program standards. This abbreviated code for identifying standards uses each of these items. For example, DGD is the Standards Reference Code for Digital Game Development. For Content Standard 2, Performance Standard 3 and Performance Indicator 4 the Standards Reference Code would be DGD.2.3.4.

CONTENT STANDARD 1.0: CONTEXT OF DIGITAL GAME DEVELOPMENT**PERFORMANCE STANDARD 1.1: UNDERSTAND HOW ADVANCES IN TECHNOLOGY IMPACT GAME DEVELOPMENT**

- 1.1.1 Explain the history of computing technologies that impact the game development industry
- 1.1.2 Explore non-digital games
- 1.1.3 Research the evolution of video games
- 1.1.4 Describe the different game genres
- 1.1.5 Evaluate contributions of individual game designers and developers

PERFORMANCE STANDARD 1.2: UNDERSTAND CAREERS IN GAME DESIGN AND DEVELOPMENT

- 1.2.1 Explore careers in designing and developing interactive experiences
- 1.2.2 Research careers in non-interactive media using real-time game engines, including Architectural, Engineering and Construction (AEC), virtual production, architectural visualization, simulation, product/industrial design, and advertising
- 1.2.3 Describe career pathways in software engineering, quality assurance, and testing
- 1.2.4 Review roles of the producer, designer, art director, tech artist, and programmer
- 1.2.5 Explain the career path of an independent developer
- 1.2.6 Research labor market information in related industries

CONTENT STANDARD 2.0: FOUNDATIONS OF GAME DESIGN AND DEVELOPMENT**PERFORMANCE STANDARD 2.1: EXPLAIN FUNDAMENTALS OF PRODUCTION**

- 2.1.1 Explain the interdependence of team members between artistic, technical, and production disciplines
- 2.1.2 Outline the process of developing a game from concept to delivery and support
- 2.1.3 Compare various types of collaboration tools, e.g., version control, shared storage, cloud services
- 2.1.4 Explain the value of version control
- 2.1.5 Explain the purpose of vertical slice
- 2.1.6 Discuss various optimization techniques
- 2.1.7 Describe good quality assurance practices

PERFORMANCE STANDARD 2.2: UNDERSTAND GAME STRUCTURE

- 2.2.1 Explore the components of game structure
- 2.2.2 Analyze the essentials of storytelling, including visual and environmental story telling
- 2.2.3 Explain the characteristics of a nonlinear story
- 2.2.4 Create rules for a game, e.g., levels and/or interactive flow
- 2.2.5 Compare conflict and outcomes
- 2.2.6 Develop objectives and outcomes for a game
- 2.2.7 Explain the importance of usability and how it impacts user experience
- 2.2.8 Explain in-game economies, motivators, and point systems

PERFORMANCE STANDARD 2.3: GAME DOCUMENTATION

- 2.3.1 Research various styles of game documentation
- 2.3.2 Develop a technical design document (TDD)
- 2.3.3 Develop components of a game design document (GDD)
- 2.3.4 Develop a list of required game assets
- 2.3.5 Produce a game design document
- 2.3.6 Produce a game pitch document
- 2.3.7 Present game documentation

PERFORMANCE STANDARD 2.4: INDUSTRY STANDARD GAME MECHANICS

- 2.4.1 Compare categories of game mechanics
- 2.4.2 Research victory condition mechanics of a game
- 2.4.3 Discuss relationships between game mechanics, game play, and interactivity
- 2.4.4 Investigate what makes a game engaging, “fun,” and playable to the user

CONTENT STANDARD 3.0: GAME DESIGN**PERFORMANCE STANDARD 3.1: UNDERSTAND FUNDAMENTALS OF DESIGN**

- 3.1.1 Evaluate the use of layout and composition
- 3.1.2 Explain color theory
- 3.1.3 Describe the principles of animation
- 3.1.4 Describe the role of perspective
- 3.1.5 Compare design considerations for environmental, assets, characters and User Interface (UI) creation
- 3.1.6 Describe the characteristics and purposes of 2D, 2.5D, and 3D art
- 3.1.7 Evaluate the importance of artistic style and implement it with continuity

PERFORMANCE STANDARD 3.2: DESIGN LEVELS

- 3.2.1 Identify components of a level and its environment
- 3.2.2 Compare processes of creating interior versus exterior environments
- 3.2.3 Compare level design of linear games to open world games
- 3.2.4 Research terrains for a specific environment
- 3.2.5 Discuss the concept of leveling up with increasing complexity
- 3.2.6 Describe the impact of story (explicit, implicit and emergent) on level design
- 3.2.7 Explain the importance of flow, distance, timing, and choke points
- 3.2.8 Storyboard levels including flow and choke points

PERFORMANCE STANDARD 3.3: DESIGN ASSETS AND CHARACTERS

- 3.3.1 Investigate the twelve major character archetypes
- 3.3.2 Contrast archetypes versus stereotypes as they relate to diversity and inclusion
- 3.3.3 Describe basic character backstory, strengths and weaknesses
- 3.3.4 Explain the visual design of characters in relation to the “game feel” and artistic style
- 3.3.5 Describe the connection between character arc and character progression
- 3.3.6 Conceptualize and illustrate original game characters and assets
- 3.3.7 Examine the roles, purpose, and design of non-player characters (NPC)
- 3.3.8 Explain the difference in design between static and dynamic game objects, e.g., props, decorations versus characters, morphing objects
- 3.3.9 Detail the difference between organic and hard surfaces

PERFORMANCE STANDARD 3.4: DESIGN CUSTOM MECHANICS

- 3.4.1 Create a victory condition
- 3.4.2 Assemble immersive elements into a game
- 3.4.3 Establish a reward system and in-game economics
- 3.4.4 Apply game mechanics to a game world
- 3.4.5 Balance and test game mechanics

CONTENT STANDARD 4.0: PROGRAMMING FOR DIGITAL GAME DEVELOPMENT**PERFORMANCE STANDARD 4.1: UNDERSTAND LOGIC IN GAME DEVELOPMENT**

- 4.1.1 Explain basic logic statements (e.g., if/then, cause/effect)
- 4.1.2 Explain the purpose and use of functions
- 4.1.3 Describe nested functions
- 4.1.4 Describe uses of Boolean operators and symbols associated with them
- 4.1.5 Demonstrate proper use of order of operations
- 4.1.6 Use logical thinking to create a diagram of code execution
- 4.1.7 Describe various types of loop structures used in programming
- 4.1.8 Describe the differences between compiled and interpreted code
- 4.1.9 Explain methods for producing artificial intelligence (AI) to control Non-Playable Characters (NPC)
- 4.1.10 Research design patterns in game programming (single, factory, and state)

PERFORMANCE STANDARD 4.2: EXPLAIN PROGRAMMING LANGUAGE CONCEPTS

- 4.2.1 Differentiate between syntax and semantics
- 4.2.2 Identify differences between compile and runtime errors
- 4.2.3 List primitive data types
- 4.2.4 Describe how arrays are used to store objects in a list
- 4.2.5 Demonstrate input from different sources
- 4.2.6 Identify expected input and output of methods/functions
- 4.2.7 Explain the connection between visual programming and coding/scripting
- 4.2.8 Compare the use of constants and variables
- 4.2.9 Describe the implications of access modifiers (private/public, local/global)
- 4.2.10 Explore engine/programming documentation to understand available methods/functions
- 4.2.11 Describe object-oriented programming (OOP)

PERFORMANCE STANDARD 4.3: UTILIZE PROGRAMMING IN GAME DEVELOPMENT

- 4.3.1 Utilize code to modify objects based on collision detection and player activation
- 4.3.2 Develop code or visual script that changes aspects of player movement (e.g., sprint, jump)
- 4.3.3 Develop code or visual script that responds to a graphic user interface (GUI) input (e.g., user interface design (UI) button press)
- 4.3.4 Develop code or visual script that responds to hardware input (e.g., keyboard key or mouse press)
- 4.3.5 Generate test cases and expected results
- 4.3.6 Format and display the value of a variable to a GUI
- 4.3.7 Implement a basic point system for a game using visual scripting or code
- 4.3.8 Verify game functionality through testing and debugging

CONTENT STANDARD 5.0: BUILD A GAME**PERFORMANCE STANDARD 5.1: EXPLORE VARIOUS DEVELOPMENT ENVIRONMENTS AND BEST PRACTICES PER PIPELINE**

- 5.1.1 Explain the characteristics and major components of game engines (render, compiler, editor)
- 5.1.2 Research various game engines for a software platform, target hardware, game style, or genre
- 5.1.3 Research supplemental tools necessary to create a specific development pipeline
- 5.1.4 Describe a complete game pipeline including the use of primary and supplemental tools and how they are used
- 5.1.5 Compare world scales between software tools
- 5.1.6 Describe considerations for importing and exporting assets
- 5.1.7 Define a standard directory structure and file naming conventions

PERFORMANCE STANDARD 5.2: DEVELOP GAME LEVELS

- 5.2.1 Define the type, structure and size of player environment
- 5.2.2 Place and define non-player characters (NPC) into the environment
- 5.2.3 Build boundaries, borders, and obstacles of levels within the game
- 5.2.4 Place triggers and develop scripted events
- 5.2.5 Create multiple levels for a game including start and closing screens and playable level(s)

PERFORMANCE STANDARD 5.3: UTILIZE GRAPHICAL USER INTERFACE (GUI)

- 5.3.1 Research examples of GUI in digital and real-world environments (e.g., Heads Up Display [HUD] and road signs)
- 5.3.2 Define and implement required elements for various GUIs
- 5.3.3 Create flowcharts that map the GUIs' functionality
- 5.3.4 Design GUIs that use standard text, 2-D, and 3-D elements (buttons, progress bars, icons, etc.)

PERFORMANCE STANDARD 5.4: USE ANIMATION IN GAME DEVELOPMENT

- 5.4.1 Create storyboards for planning animations
- 5.4.2 Change an object's state or position over time
- 5.4.3 Establish an object's relative speed (timing versus frame rate)
- 5.4.4 Describe the difference between global time and local time
- 5.4.5 Describe the difference between forward and inverse kinematics
- 5.4.6 Examine the process of particle creation and its application to game design
- 5.4.7 Explain how joints, sockets, bones, and skins are used
- 5.4.8 Create a parent/child hierarchy
- 5.4.9 Simulate rigid body dynamics (e.g., shattering wall, breaking glass)
- 5.4.10 Animate game objects using triggers
- 5.4.11 Describe the process of motion capture for animation
- 5.4.12 Practice the use of cinematics and camera movements in game engines

PERFORMANCE STANDARD 5.5: INTEGRATE VARIOUS MEDIA TYPES

- 5.5.1 Integrate different types of audio (e.g., sound effects, ambient background, dialogue, user experience design (UX), and score)
- 5.5.2 Practice creating triggers for sound, mixes, and/or loops
- 5.5.3 Determine acceptable media formats and files for game development (e.g., sound, graphics, video)
- 5.5.4 Identify and import appropriate media into a game engine

PERFORMANCE STANDARD 5.6: CREATE GAME ART

- 5.6.1 Explain the application of low polygon and high polygon construction for real-time environments
- 5.6.2 Create game assets from references and game design documents (GDD)
- 5.6.3 Create hard surface and organic models using design principles (props or characters)
- 5.6.4 Apply texturing/surfacing/shading/grunging to models and normal mapping
- 5.6.5 Render or produce a high-resolution screenshot from within a game engine
- 5.6.6 Real-time render a video, animated sequence or cutscene in a game engine
- 5.6.7 Differentiate UVW mapping and world coordinate systems
- 5.6.8 Explain the importance of pre-baking lights and shadows for real-time interactive environments
- 5.6.9 Implement basic lighting for ambient and artificial light
- 5.6.10 Create an environment including terrain, foliage and/or static objects
- 5.6.11 Create custom materials/shaders
- 5.6.12 Describe the use of Level of Detail (LOD) substitution, MIP mapping textures, and culling objects

CONTENT STANDARD 6.0: LEGAL AND ETHICAL ISSUES IN GAME DESIGN AND DEVELOPMENT**PERFORMANCE STANDARD 6.1: UNDERSTAND LEGAL CONSIDERATIONS IN GAME DEVELOPMENT**

- 6.1.1 Research laws that govern intellectual property in diverse forms
- 6.1.2 Evaluate Creative Commons and open source licensure
- 6.1.3 Explain copyright, trademark, and other intellectual property protection
- 6.1.4 Identify key elements of non-disclosure agreements (NDA) and contracts
- 6.1.5 Adhere to intellectual property laws and regulations and cite proprietary content and derivative works
- 6.1.6 Explain Entertainment Software Rating Board (ESRB) and other rating systems for digital games

PERFORMANCE STANDARD 6.2: EXPLAIN SECURITY ISSUES IN RELATION TO GAME DEVELOPMENT AND DESIGN

- 6.2.1 Explain invasion of privacy in the use of technology
- 6.2.2 Explore the issues of piracy and digital rights management (DRM)
- 6.2.3 Model acceptable security practices

PERFORMANCE STANDARD 6.3: UNDERSTAND THE IMPORTANCE OF ETHICS, DIVERSITY, AND INCLUSION

- 6.3.1 Discuss diversity and inclusivity in games and the gaming industry
- 6.3.2 Discuss social responsibility and issues concerning video gaming

CONTENT STANDARD 7.0: ADVANCED TOPICS IN DIGITAL GAME DEVELOPMENT**PERFORMANCE STANDARD 7.1: UNDERSTAND SOCIAL ASPECTS OF GAMING**

- 7.1.1 Describe integration of social components in a game
- 7.1.2 Explain the role of social media in the gaming community
- 7.1.3 Describe professional events in digital gaming

PERFORMANCE STANDARD 7.2: UNDERSTAND THE ROLE OF NETWORKING

- 7.2.1 Summarize characteristics of cloud gaming
- 7.2.2 Evaluate the advances of multi-player gaming

PERFORMANCE STANDARD 7.3: EXPLORE ADVANCES IN DEVICES

- 7.3.1 Discuss trends in input devices
- 7.3.2 Examine current trends in output devices and displays
- 7.3.3 Explore advances in peripheral devices