

Proposal for a Student-Defined Minor in Video Game Development

Draft #4

Name: Duncan Boehle

Current Class: Sophomore

Current College/Major: Computer Science

Date: 2/21/10

Introduction:

I would like to pursue a minor in video game development, as an interdisciplinary minor that holistically studies both the art and science of making video games. These two perspectives encompass the following fields: visual art, narrative, human-computer interaction, psychology, software engineering, group management, and graphics. No other programs in the university can both cover such diverse fields of study and strongly tie them together with a single focus.

Description and Rationale:

Although video games are fundamentally computer programs, their creation consists of far more than a computer science-based program could teach. Any one member of a game development project must communicate with many different kinds of engineers, artists, and producers; understanding these fields is extremely important for every member of the project. The combination of specializing in one field while collaborating with so many different people lends itself to the creation of a minor in video game development, with a student's major being the actual work focus.

One invaluable aspect of video games is their narrative; writers carefully create believable characters, settings, and a deep history for the game's environment. Because the player does not passively view the narrative, as in many other works of art, writers must ensure that the player can interact with, modify,

and spend time with characters in a simulated environment. These additional characters give writers the extra challenge of creating dozens of side quests, back stories, and game lore that is consistent with the central narrative. Understanding this writing process is essential for all members of a game project who deal with characters.

Due to the interactive nature of these characters, studying psychology is extremely important for creating an immersive environment. There are two main applications of psychology with video games: emulating human psychology in non-player-characters (NPCs), and understanding how the player thinks and will interact with the game. The former application of psychology is partially covered by writers, who must explore the motivations and purposes of the game's characters to determine how characters act on a narrative level. The field of artificial intelligence, however, is necessary for determining how characters act on a physical level. Generating realistic movement and animation in dynamic environments, working together to achieve a complex goal, and interpreting the player's actions are all possible areas of study under artificial intelligence.

Understanding and reacting to the player's actions, however, are not the responsibilities of only the in-game characters. A fundamental aspect of software design is predicting how users will interact with the interface, what their intentions are, and how they will feel during certain behavior of the software. These principles must be applied to the user interface of a game, the game's controls/play mechanics, and the player's involvement in a dynamic narrative. Video games' interactivity makes them an extremely unique form of art; all

members of a game development project must rigorously understand human-computer interaction in order to create something that is both a coherent art form and a useable piece of software.

The creation of the software itself is one of the most important aspects of video game development. Unlike other forms of games, which can take the form of any interactive medium, video games require a modular, efficient, portable, and flexible code framework supporting them. Creating this type of architecture would require not only the programming experience granted from the Computer Science major, but also specialized knowledge of graphics, networking, artificial intelligence, and physics simulation. For non-programmers, however, at least a basic understanding of programming paradigms is necessary in order to create content. Artists, managers, designers, musicians, and writers for video games should all know what kinds of content are compatible with the game's framework, and they must be able to work in a constrained environment. For example, artists and musicians for video games of the late 80's were able to create legendary soundtracks and characters using only 3 instruments and 53 colors. Today, the rapid growth of 3D technologies means that artists need a firm understanding of the consequences of choosing certain modeling applications, file formats, or animation methods. Although technical proficiency will vary drastically across a game development team, a basic familiarity with computer science is necessary for effective collaboration.

Because programming is such an essential and laborious aspect of game development, project workflows are often very similar to that of large, traditional

pieces of software. The collaboration between software engineers and artists creates a unique perspective, however; programmers are creating the framework and medium through which art is expressed. Developers, then, need to adapt the typical software development cycle to the creative process and workflow of artistic projects, in order to create a game as a work of art instead of just a technical demo. An academic study of art focuses on many aspects of this creative process: concept iteration, shaping implementation around purpose, and abstracting from individual details to user experience, from emotional reactions to social implications. Every member of a game development project is an essential part of this creative process at every step of the way, and working towards a cohesive artistic expression is only possible with a shared mindset. The common notion of a vast disparity between engineering and art simply cannot exist in a strong group, and gaining first-hand experience in the creative process is the best way to start to combine the two schools of thought.

The purpose of studying several other disciplines is fundamentally to create an environment where group members can collaborate on a basis of understanding and a cohesive goal, rather than a forced union of opposing ideas. The fundamental techniques of working in a group and understanding leadership apply regardless of discipline, so a general study of organizational mechanics and working in groups is essential for every member of a project. Equipped with both a general understanding of how to work in a group and a true understanding of different creative perspectives, game developers can succeed to create both an extraordinary technological feat and an interactive artistic masterpiece.

Video Game Development Curriculum: Total: 69-84 units

Narrative and Writing Total: 9 units

One of the following:

76-260 Survey of Forms: Fiction 9 units

76-269 Survey of Forms: Screenwriting 9 units

Psychology Total: 9 units

One of the following:

85-211 Cognitive Psychology 9 units

80-213 Human Information Processing and A.I. 9 units

Programming and Human-Computer Interaction Total: 22 units

15-101/15-102/15-110 Introduction to Programming 10 units

Plus one of the following:

05-391 Designing Human Centered Software 12 units

05-430/05-431 User Interface Programming 12 units

15-466 Computer Game Programming 12 units

Art Total: 20 units

60-424 Adv. ETB: Game Design for Artists 10 units

Plus one of the following:

60-110 Electronic Media Studio 10 units

60-210 Electronic Media Studio II 10 units

60-415 Advanced ETB: Animation 10 units

60-416 Advanced ETB: Interactive Multimedia 10 units

Interdisciplinary Development Total: 9-24 units

One of the following:

70-311 Organizational Behavior 9 units

15-465/60-414 Animation Art and Technology 12 units

53-608 Visual Storytelling 12 units

53-609 Game Design 12 units

53-831 Building Virtual Worlds 24 units